Package ‘paws’

May 16, 2024

Title Amazon Web Services Software Development Kit

Version 0.6.0

Description Interface to Amazon Web Services <https://aws.amazon.com>, including storage, database, and compute services, such as 'Simple Storage Service' ('S3'), 'DynamoDB' 'NoSQL' database, and 'Lambda' functions-as-a-service.

License Apache License (>= 2.0)

URL https://github.com/paws-r/paws

BugReports https://github.com/paws-r/paws/issues

Imports paws.analytics (>= 0.6.0), paws.application.integration (>= 0.6.0), paws.common (>= 0.6.0), paws.compute (>= 0.6.0), paws.cost.management (>= 0.6.0), paws.customer.engagement (>= 0.6.0), paws.database (>= 0.6.0), paws.developer.tools (>= 0.6.0), paws.end.user.computing (>= 0.6.0), paws.machine.learning (>= 0.6.0), paws.management (>= 0.6.0), paws.networking (>= 0.6.0), paws.security.identity (>= 0.6.0), paws.storage (>= 0.6.0)

Suggests testthat

Encoding UTF-8

RoxygenNote 7.3.1

NeedsCompilation no

Author David Kretch [aut], Adam Banker [aut], Dyfan Jones [cre], Amazon.com, Inc. [cph]

Maintainer Dyfan Jones <dyfan.r.jones@gmail.com>

Repository CRAN

Date/Publication 2024-05-16 16:40:02 UTC
R topics documented:

accessanalyzer .............................................................. 8
account ................................................................. 11
acm ................................................................. 13
acmpca ................................................................. 16
apigateway ............................................................. 19
apigatewaymanagementapi ........................................... 23
apigatewayv2 ........................................................... 26
appfabric ............................................................... 30
applicationautoscaling .................................................... 32
applicationcostprofiler .................................................. 36
applicationinsights ...................................................... 38
appmesh ............................................................... 41
appregistry ............................................................. 44
apprunner .............................................................. 47
appstream ............................................................. 51
arczonalshift ........................................................... 55
athena ................................................................. 58
auditmanager ............................................................ 61
augmentedairuntime ..................................................... 66
autoscaling ............................................................... 69
autoscalingplans ......................................................... 73
backup ................................................................. 75
backupgateway ........................................................... 79
backupstorage ............................................................ 82
batch ................................................................. 85
bedrock ............................................................... 88
bedrockruntime ........................................................ 90
billingconductor ........................................................ 93
braket ................................................................. 96
budgets ............................................................... 98
cloud9 ............................................................... 101
cloudcontrolapi .......................................................... 105
clouddirectory ........................................................... 107
cloudformation ........................................................ 111
cloudfront ........................................................... 115
cloudhsm ............................................................. 120
cloudhsmv2 ............................................................ 123
cloudsearch ........................................................... 125
cloudsearchdomain .................................................... 128
cloudtrail ........................................................... 131
cloudtraildataservice .................................................. 134
cloudwatch ........................................................ 136
cloudwatchevents ....................................................... 140
cloudwatchevidently .................................................... 143
cloudwatchinternetmonitor ............................................ 146
cloudwatchlogs ......................................................... 149
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>cloudwatchobservabilityaccessmanager</td>
<td>154</td>
</tr>
<tr>
<td>cloudwatchrum</td>
<td>157</td>
</tr>
<tr>
<td>codeartifact</td>
<td>159</td>
</tr>
<tr>
<td>codebuild</td>
<td>166</td>
</tr>
<tr>
<td>codecatalyst</td>
<td>169</td>
</tr>
<tr>
<td>codecommit</td>
<td>174</td>
</tr>
<tr>
<td>codedeploy</td>
<td>181</td>
</tr>
<tr>
<td>codeguruprofiler</td>
<td>186</td>
</tr>
<tr>
<td>codegurureviewer</td>
<td>189</td>
</tr>
<tr>
<td>codegurusecurity</td>
<td>191</td>
</tr>
<tr>
<td>codepipeline</td>
<td>194</td>
</tr>
<tr>
<td>codestar</td>
<td>199</td>
</tr>
<tr>
<td>codestarconnections</td>
<td>202</td>
</tr>
<tr>
<td>codestarnotifications</td>
<td>206</td>
</tr>
<tr>
<td>cognitoidentity</td>
<td>209</td>
</tr>
<tr>
<td>cognitoidentityprovider</td>
<td>212</td>
</tr>
<tr>
<td>cognitosync</td>
<td>217</td>
</tr>
<tr>
<td>comprehend</td>
<td>220</td>
</tr>
<tr>
<td>comprehendmedical</td>
<td>224</td>
</tr>
<tr>
<td>computeoptimizer</td>
<td>227</td>
</tr>
<tr>
<td>configservice</td>
<td>230</td>
</tr>
<tr>
<td>connect</td>
<td>234</td>
</tr>
<tr>
<td>connectcampaignservice</td>
<td>242</td>
</tr>
<tr>
<td>connectcases</td>
<td>244</td>
</tr>
<tr>
<td>connectcontactlens</td>
<td>247</td>
</tr>
<tr>
<td>connectparticipant</td>
<td>250</td>
</tr>
<tr>
<td>connectwisdomservice</td>
<td>252</td>
</tr>
<tr>
<td>controltower</td>
<td>255</td>
</tr>
<tr>
<td>costandusagereportservice</td>
<td>259</td>
</tr>
<tr>
<td>costexplorer</td>
<td>262</td>
</tr>
<tr>
<td>customerprofiles</td>
<td>265</td>
</tr>
<tr>
<td>datapipeline</td>
<td>268</td>
</tr>
<tr>
<td>datazone</td>
<td>271</td>
</tr>
<tr>
<td>dax</td>
<td>276</td>
</tr>
<tr>
<td>detective</td>
<td>279</td>
</tr>
<tr>
<td>devopsnguru</td>
<td>282</td>
</tr>
<tr>
<td>directconnect</td>
<td>286</td>
</tr>
<tr>
<td>directoryservice</td>
<td>289</td>
</tr>
<tr>
<td>dlm</td>
<td>293</td>
</tr>
<tr>
<td>docdb</td>
<td>295</td>
</tr>
<tr>
<td>docdbelastic</td>
<td>299</td>
</tr>
<tr>
<td>drs</td>
<td>301</td>
</tr>
<tr>
<td>dynamodb</td>
<td>305</td>
</tr>
<tr>
<td>dynamodbstreams</td>
<td>309</td>
</tr>
<tr>
<td>ebs</td>
<td>311</td>
</tr>
<tr>
<td>ec2</td>
<td>314</td>
</tr>
<tr>
<td>ec2instanceconnect</td>
<td>329</td>
</tr>
<tr>
<td>ecr</td>
<td>331</td>
</tr>
<tr>
<td>Topic</td>
<td>Page</td>
</tr>
<tr>
<td>--------------------------</td>
<td>------</td>
</tr>
<tr>
<td>ecrpublic</td>
<td>335</td>
</tr>
<tr>
<td>ecs</td>
<td>337</td>
</tr>
<tr>
<td>efs</td>
<td>341</td>
</tr>
<tr>
<td>eks</td>
<td>344</td>
</tr>
<tr>
<td>elasticache</td>
<td>348</td>
</tr>
<tr>
<td>elasticbeanstalk</td>
<td>352</td>
</tr>
<tr>
<td>elasticinference</td>
<td>355</td>
</tr>
<tr>
<td>elasticsearchservice</td>
<td>358</td>
</tr>
<tr>
<td>elb</td>
<td>361</td>
</tr>
<tr>
<td>elbv2</td>
<td>364</td>
</tr>
<tr>
<td>emr</td>
<td>368</td>
</tr>
<tr>
<td>emrcontainers</td>
<td>371</td>
</tr>
<tr>
<td>emrserverless</td>
<td>374</td>
</tr>
<tr>
<td>entityresolution</td>
<td>377</td>
</tr>
<tr>
<td>eventbridge</td>
<td>380</td>
</tr>
<tr>
<td>eventbridgepipes</td>
<td>384</td>
</tr>
<tr>
<td>eventbridgescheduler</td>
<td>386</td>
</tr>
<tr>
<td>finspace</td>
<td>389</td>
</tr>
<tr>
<td>finspacedata</td>
<td>392</td>
</tr>
<tr>
<td>firehose</td>
<td>395</td>
</tr>
<tr>
<td>fis</td>
<td>397</td>
</tr>
<tr>
<td>fms</td>
<td>400</td>
</tr>
<tr>
<td>forecastqueryservice</td>
<td>403</td>
</tr>
<tr>
<td>forecastservice</td>
<td>405</td>
</tr>
<tr>
<td>frauddetector</td>
<td>409</td>
</tr>
<tr>
<td>fsx</td>
<td>413</td>
</tr>
<tr>
<td>glacier</td>
<td>416</td>
</tr>
<tr>
<td>globalaccelerator</td>
<td>419</td>
</tr>
<tr>
<td>glue</td>
<td>424</td>
</tr>
<tr>
<td>gluedatabrew</td>
<td>430</td>
</tr>
<tr>
<td>guardduty</td>
<td>433</td>
</tr>
<tr>
<td>health</td>
<td>437</td>
</tr>
<tr>
<td>healthlake</td>
<td>441</td>
</tr>
<tr>
<td>iam</td>
<td>443</td>
</tr>
<tr>
<td>iamrolesanywhere</td>
<td>449</td>
</tr>
<tr>
<td>identitystore</td>
<td>452</td>
</tr>
<tr>
<td>imagebuilder</td>
<td>455</td>
</tr>
<tr>
<td>inspector</td>
<td>458</td>
</tr>
<tr>
<td>inspector2</td>
<td>462</td>
</tr>
<tr>
<td>ivs</td>
<td>465</td>
</tr>
<tr>
<td>ivschat</td>
<td>471</td>
</tr>
<tr>
<td>ivsrealt ime</td>
<td>476</td>
</tr>
<tr>
<td>kafka</td>
<td>481</td>
</tr>
<tr>
<td>kafkaconnect</td>
<td>484</td>
</tr>
<tr>
<td>kendra</td>
<td>487</td>
</tr>
<tr>
<td>kendraranking</td>
<td>490</td>
</tr>
<tr>
<td>keyspaces</td>
<td>492</td>
</tr>
<tr>
<td>kinesis</td>
<td>495</td>
</tr>
</tbody>
</table>
topics documented:

- kinesisanalytics .......................... 498
- kinesisanalyticsv2 ......................... 501
- kns ........................................ 504
- lakeformation .............................. 508
- lambda .................................. 512
- lexmodelbuildingservice .................. 516
- lexmodelsv2 ............................... 519
- lexruntimeservice ......................... 524
- lexruntimev2 ............................... 526
- licensemanager ............................. 529
- licensemanagerlinuxsubscriptions ........ 532
- licensemanagerusersubscriptions ........ 534
- lightsail .................................. 537
- locationservice ........................... 543
- lookoutequipment ......................... 546
- lookoutmetrics ............................ 549
- machinelearning ........................... 552
- macie2 ................................... 555
- managedgrafana ............................. 559
- marketplacecatalog ....................... 562
- marketplacecommerceanalytics .......... 565
- marketplaceentitlementservice .......... 567
- marketplacemetering ...................... 570
- memorydb .................................. 573
- mq ........................................ 576
- mturk ..................................... 578
- mwaa ...................................... 581
- neptune ................................... 584
- neptunedata ............................... 588
- networkfirewall ............................ 591
- networkmanager ............................ 595
- nimblestudio ................................ 599
- omics ...................................... 602
- opensearchingestion ....................... 606
- opensearchservice ......................... 609
- opensearchserviceserverless ............. 612
- opsworks .................................. 616
- opworkscm ................................ 620
- organizations .............................. 624
- panorama .................................. 628
- paymentcryptographycontrolplane ....... 631
- paymentcryptographydataplane ........... 634
- pcaconnectorad ............................ 637
- personalize ............................... 639
- personalizeevents ......................... 643
- personalizeruntime ......................... 645
- pi ......................................... 648
- pinpoint ................................... 651
R topics documented:

pinpointemail ............................................................. 655
pinpointsmsvoice ........................................................... 659
pinpointsmsvoicev2 ......................................................... 661
polly ................................................................. 666
pricing ................................................................. 668
prometheusservice ......................................................... 671
proton ................................................................. 674
qldb ................................................................. 680
qldbsession ............................................................... 682
quicksight ............................................................... 685
ram ................................................................. 691
rds ................................................................. 694
rdsdataservice ............................................................. 701
recyclebin ............................................................... 703
redshift ................................................................. 706
redshiftdataapiservice .................................................. 711
redshiftserverless ........................................................ 714
rekognition ............................................................... 717
resiliencehub ............................................................. 723
resourceexplorer .......................................................... 727
resourcegroups ........................................................... 730
resourcegroupstaggingapi .............................................. 733
route53 ................................................................. 735
route53domains ............................................................ 739
route53recoverycluster .................................................. 742
route53recoverycontrolconfig ........................................ 745
route53recoveryreadiness ............................................... 748
route53resolver ........................................................... 751
s3 ................................................................. 755
s3control ................................................................. 759
s3outposts ................................................................. 763
sagemaker ................................................................. 766
sagemakeredgemanager .................................................. 776
sagemakerfeaturestoreruntime ......................................... 778
sagemakergeospatialcapabilities ...................................... 780
sagemakermetrics .......................................................... 783
sagemakerruntime ........................................................ 785
savingsplans ................................................................. 788
schemas ................................................................. 790
secretsmanager ............................................................ 793
securityhub ............................................................... 796
securitylake ............................................................... 801
serverlessapplicationrepository ...................................... 805
servicecatalog ............................................................. 808
servicediscovery .......................................................... 812
servicequotas .............................................................. 815
ses ................................................................. 818
sesv2 ................................................................. 821
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>sfn</td>
<td>826</td>
</tr>
<tr>
<td>shield</td>
<td>829</td>
</tr>
<tr>
<td>simpledb</td>
<td>832</td>
</tr>
<tr>
<td>sns</td>
<td>834</td>
</tr>
<tr>
<td>sqs</td>
<td>838</td>
</tr>
<tr>
<td>ssm</td>
<td>841</td>
</tr>
<tr>
<td>ssmcontacts</td>
<td>846</td>
</tr>
<tr>
<td>ssmincidents</td>
<td>849</td>
</tr>
<tr>
<td>ssmmap</td>
<td>852</td>
</tr>
<tr>
<td>sso</td>
<td>855</td>
</tr>
<tr>
<td>ssoadmin</td>
<td>857</td>
</tr>
<tr>
<td>ssooidc</td>
<td>861</td>
</tr>
<tr>
<td>storagegateway</td>
<td>864</td>
</tr>
<tr>
<td>sts</td>
<td>869</td>
</tr>
<tr>
<td>support</td>
<td>872</td>
</tr>
<tr>
<td>supportapp</td>
<td>875</td>
</tr>
<tr>
<td>swf</td>
<td>878</td>
</tr>
<tr>
<td>synthetics</td>
<td>881</td>
</tr>
<tr>
<td>telconetworkbuilder</td>
<td>884</td>
</tr>
<tr>
<td>textract</td>
<td>887</td>
</tr>
<tr>
<td>timestreamquery</td>
<td>890</td>
</tr>
<tr>
<td>timestreamwrite</td>
<td>892</td>
</tr>
<tr>
<td>transcribeservice</td>
<td>895</td>
</tr>
<tr>
<td>translate</td>
<td>898</td>
</tr>
<tr>
<td>verifiedpermissions</td>
<td>901</td>
</tr>
<tr>
<td>voiceid</td>
<td>905</td>
</tr>
<tr>
<td>vpclattice</td>
<td>907</td>
</tr>
<tr>
<td>waf</td>
<td>911</td>
</tr>
<tr>
<td>wafregional</td>
<td>915</td>
</tr>
<tr>
<td>wafv2</td>
<td>919</td>
</tr>
<tr>
<td>wellarchitected</td>
<td>923</td>
</tr>
<tr>
<td>workdocs</td>
<td>927</td>
</tr>
<tr>
<td>worklink</td>
<td>930</td>
</tr>
<tr>
<td>workmail</td>
<td>933</td>
</tr>
<tr>
<td>workmailmessageflow</td>
<td>937</td>
</tr>
<tr>
<td>workspaces</td>
<td>940</td>
</tr>
<tr>
<td>workspacesweb</td>
<td>944</td>
</tr>
<tr>
<td>xray</td>
<td>947</td>
</tr>
</tbody>
</table>
Description

Identity and Access Management Access Analyzer helps you to set, verify, and refine your IAM policies by providing a suite of capabilities. Its features include findings for external and unused access, basic and custom policy checks for validating policies, and policy generation to generate fine-grained policies. To start using IAM Access Analyzer to identify external or unused access, you first need to create an analyzer.

**External access analyzers** help identify potential risks of accessing resources by enabling you to identify any resource policies that grant access to an external principal. It does this by using logic-based reasoning to analyze resource-based policies in your Amazon Web Services environment. An external principal can be another Amazon Web Services account, a root user, an IAM user or role, a federated user, an Amazon Web Services service, or an anonymous user. You can also use IAM Access Analyzer to preview public and cross-account access to your resources before deploying permissions changes.

**Unused access analyzers** help identify potential identity access risks by enabling you to identify unused IAM roles, unused access keys, unused console passwords, and IAM principals with unused service and action-level permissions.

Beyond findings, IAM Access Analyzer provides basic and custom policy checks to validate IAM policies before deploying permissions changes. You can use policy generation to refine permissions by attaching a policy generated using access activity logged in CloudTrail logs.

This guide describes the IAM Access Analyzer operations that you can call programmatically. For general information about IAM Access Analyzer, see Identity and Access Management Access Analyzer in the IAM User Guide.

Usage

```python
accessanalyzer(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

**Arguments**

- `config`: Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
    - `creds`:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint**: The complete URL to use for the constructed client.

**region**: The AWS Region used in instantiating the client.

**close_connection**: Immediately close all HTTP connections.

**timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

**s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.

**sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

**credentials** Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

### Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

### Service syntax

```r
csvc <- accessanalyzer(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
  ),
```
```python
    timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
    credentials = list(
        creds = list(
            access_key_id = "string",
            secret_access_key = "string",
            session_token = "string"
        ),
        profile = "string",
        anonymous = "logical"
    ),
    endpoint = "string",
    region = "string"
)

Operations

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>apply_archive_rule</td>
<td>Retroactively applies the archive rule to existing findings that meet the archive rule criteria</td>
</tr>
<tr>
<td>cancel_policy_generation</td>
<td>Cancels the requested policy generation</td>
</tr>
<tr>
<td>check_access_not_granted</td>
<td>Checks whether the specified access isn’t allowed by a policy</td>
</tr>
<tr>
<td>check_no_new_access</td>
<td>Checks whether new access is allowed for an updated policy when compared to the existing policy</td>
</tr>
<tr>
<td>create_access_preview</td>
<td>Creates an access preview that allows you to preview IAM Access Analyzer findings for your resource</td>
</tr>
<tr>
<td>create_analyzer</td>
<td>Creates an analyzer for your account</td>
</tr>
<tr>
<td>create_archive_rule</td>
<td>Creates an archive rule for the specified analyzer</td>
</tr>
<tr>
<td>delete_analyzer</td>
<td>Deletes the specified analyzer</td>
</tr>
<tr>
<td>delete_archive_rule</td>
<td>Deletes the specified archive rule</td>
</tr>
<tr>
<td>get_access_preview</td>
<td>Retrieves information about an access preview for the specified analyzer</td>
</tr>
<tr>
<td>get_analyzed_resource</td>
<td>Retrieves information about a resource that was analyzed</td>
</tr>
<tr>
<td>get_analyzer</td>
<td>Retrieves information about the specified analyzer</td>
</tr>
<tr>
<td>get_archive_rule</td>
<td>Retrieves information about an archive rule</td>
</tr>
<tr>
<td>get_finding</td>
<td>Retrieves information about the specified finding</td>
</tr>
<tr>
<td>get_finding_v2</td>
<td>Retrieves information about the specified finding</td>
</tr>
<tr>
<td>get_generated_policy</td>
<td>Retrieves the policy that was generated using StartPolicyGeneration</td>
</tr>
<tr>
<td>list_access_preview_findings</td>
<td>Retrieves a list of access preview findings generated by the specified access preview</td>
</tr>
<tr>
<td>list_access_previews</td>
<td>Retrieves a list of access previews for the specified analyzer</td>
</tr>
<tr>
<td>list_analyzed_resources</td>
<td>Retrieves a list of resources of the specified type that have been analyzed by the specified external access analyzer</td>
</tr>
<tr>
<td>list_analyzers</td>
<td>Retrieves a list of analyzers</td>
</tr>
<tr>
<td>list_archive_rules</td>
<td>Retrieves a list of archive rules created for the specified analyzer</td>
</tr>
<tr>
<td>list_findings</td>
<td>Retrieves a list of findings generated by the specified analyzer</td>
</tr>
<tr>
<td>list_findings_v2</td>
<td>Retrieves a list of findings generated by the specified analyzer</td>
</tr>
<tr>
<td>list_policy_generations</td>
<td>Lists all of the policy generations requested in the last seven days</td>
</tr>
<tr>
<td>list_tags_for_resource</td>
<td>Retrieves a list of tags applied to the specified resource</td>
</tr>
<tr>
<td>start_policy_generation</td>
<td>Starts the policy generation request</td>
</tr>
<tr>
<td>start_resource_scan</td>
<td>Immediately starts a scan of the policies applied to the specified resource</td>
</tr>
<tr>
<td>tag_resource</td>
<td>Adds a tag to the specified resource</td>
</tr>
<tr>
<td>untag_resource</td>
<td>Removes a tag from the specified resource</td>
</tr>
</tbody>
</table>
account

update_archive_rule
update_findings
validate_policy

Updates the criteria and values for the specified archive rule
Updates the status for the specified findings
Requests the validation of a policy and returns a list of findings

Examples

## Not run:
svc <- accessanalyzer()
svc$apply_archive_rule(
  Foo = 123
)

## End(Not run)

account

<table>
<thead>
<tr>
<th>AWS Account</th>
</tr>
</thead>
</table>

Description

Operations for Amazon Web Services Account Management

Usage

account(config = list(), credentials = list(), endpoint = NULL, region = NULL)

Arguments

<table>
<thead>
<tr>
<th>config</th>
<th>Optional configuration of credentials, endpoint, and/or region.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• credentials:</td>
</tr>
<tr>
<td></td>
<td>• creds:</td>
</tr>
<tr>
<td></td>
<td>• access_key_id: AWS access key ID</td>
</tr>
<tr>
<td></td>
<td>• secret_access_key: AWS secret access key</td>
</tr>
<tr>
<td></td>
<td>• session_token: AWS temporary session token</td>
</tr>
<tr>
<td></td>
<td>• profile: The name of a profile to use. If not given, then the default profile is used.</td>
</tr>
<tr>
<td></td>
<td>• anonymous: Set anonymous credentials.</td>
</tr>
<tr>
<td></td>
<td>• endpoint: The complete URL to use for the constructed client.</td>
</tr>
<tr>
<td></td>
<td>• region: The AWS Region used in instantiating the client.</td>
</tr>
<tr>
<td></td>
<td>• close_connection: Immediately close all HTTP connections.</td>
</tr>
<tr>
<td></td>
<td>• timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.</td>
</tr>
<tr>
<td></td>
<td>• s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. <a href="http://s3.amazonaws.com/BUCKET/KEY">http://s3.amazonaws.com/BUCKET/KEY</a>.</td>
</tr>
</tbody>
</table>
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials**
Optional credentials shorthand for the config parameter

• **creds**:
  – **access_key_id**: AWS access key ID
  – **secret_access_key**: AWS secret access key
  – **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- account(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"),
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"),
      profile = "string",
    )
  )
)```

acm

anonymous = "logical",
endpoint = "string",
region = "string"

Operations

delete_alternate_contact Deletes the specified alternate contact from an Amazon Web Services account
disable_region Disables (opts-out) a particular Region for an account
enable_region Enables (opts-in) a particular Region for an account
get_alternate_contact Retrieves the specified alternate contact attached to an Amazon Web Services account
get_contact_information Retrieves the primary contact information of an Amazon Web Services account
get_region_opt_status Retrieves the opt-in status of a particular Region
list_regions Lists all the Regions for a given account and their respective opt-in statuses
put_alternate_contact Modifies the specified alternate contact attached to an Amazon Web Services account
put_contact_information Updates the primary contact information of an Amazon Web Services account

Examples

## Not run:
svc <- account()
svc$delete_alternate_contact(
  Foo = 123
)

## End(Not run)

---

**acm**

**AWS Certificate Manager**

**Description**

Certificate Manager

You can use Certificate Manager (ACM) to manage SSL/TLS certificates for your Amazon Web Services-based websites and applications. For more information about using ACM, see the Certificate Manager User Guide.

**Usage**

acm(config = list(), credentials = list(), endpoint = NULL, region = NULL)
Arguments

config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

svc <- acm(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
...
acm

secret_access_key = "string",
session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string",
close_connection = "logical",
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

add_tags_to_certificate Adds one or more tags to an ACM certificate
delete_certificate Deletes a certificate and its associated private key
describe_certificate Returns detailed metadata about the specified ACM certificate
export_certificate Exports a private certificate issued by a private certificate authority (CA) for use anywhere
get_account_configuration Returns the account configuration options associated with an Amazon Web Services account
get_certificate Retrieves an Amazon-issued certificate and its certificate chain
import_certificate Imports a certificate into Certificate Manager (ACM) to use with services that are integrated with ACM
list_certificates Retrieves a list of certificate ARNs and domain names
list_tags_for_certificate Lists the tags that have been applied to the ACM certificate
put_account_configuration Adds or modifies account-level configurations in ACM
remove_tags_from_certificate Remove one or more tags from an ACM certificate
renew_certificate Renews an eligible ACM certificate
request_certificate Requests an ACM certificate for use with other Amazon Web Services services
resend_validation_email Resends the email that requests domain ownership validation
update_certificate_options Updates a certificate
Examples

```r
## Not run:
svc <- acm()
svc$add_tags_to_certificate(
  Foo = 123
)

## End(Not run)
```

---

**Description**

This is the Amazon Web Services Private Certificate Authority API Reference. It provides descriptions, syntax, and usage examples for each of the actions and data types involved in creating and managing a private certificate authority (CA) for your organization.

The documentation for each action shows the API request parameters and the JSON response. Alternatively, you can use one of the Amazon Web Services SDKs to access an API that is tailored to the programming language or platform that you prefer. For more information, see Amazon Web Services SDKs.

Each Amazon Web Services Private CA API operation has a quota that determines the number of times the operation can be called per second. Amazon Web Services Private CA throttles API requests at different rates depending on the operation. Throttling means that Amazon Web Services Private CA rejects an otherwise valid request because the request exceeds the operation's quota for the number of requests per second. When a request is throttled, Amazon Web Services Private CA returns a ThrottlingException error. Amazon Web Services Private CA does not guarantee a minimum request rate for APIs.

To see an up-to-date list of your Amazon Web Services Private CA quotas, or to request a quota increase, log into your Amazon Web Services account and visit the Service Quotas console.

**Usage**

```r
acmpca(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

**Arguments**

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
– **anonymous**: Set anonymous credentials.

- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

**credentials**
Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- acmpca(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
  )
)
```
stsRegionalEndpoint = "string"
),
credentials = list(
    creds = list(
        accessKeyId = "string",
        secretAccessKey = "string",
        sessionToken = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

create_certificate_authority
create_certificate_authority_audit_report
create_permission
delete_certificate_authority
delete_permission
delete_policy
describe_certificate_authority
describe_certificate_authority_audit_report
get_certificate
get_certificate_authority_certificate
get_certificate_authority_csr
get_policy
import_certificate_authority_certificate
issue_certificate
list_certificate Authorities
list_permissions
list_tags
put_policy
restore_certificate_authority
revoke_certificate
tag_certificate_authority
untag_certificate_authority
update_certificate_authority

Examples

## Not run:
svc <- acmpca()
svc$create_certificate_authority(
Amazon API Gateway helps developers deliver robust, secure, and scalable mobile and web application back ends. API Gateway allows developers to securely connect mobile and web applications to APIs that run on Lambda, Amazon EC2, or other publicly addressable web services that are hosted outside of AWS.

Usage

```r
apigateway(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)
credentials: Optional credentials shorthand for the config parameter

- creds:
  - access_key_id: AWS access key ID
  - secret_access_key: AWS secret access key
  - session_token: AWS temporary session token

- profile: The name of a profile to use. If not given, then the default profile is used.

- anonymous: Set anonymous credentials.

endpoint: Optional shorthand for complete URL to use for the constructed client.

region: Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like 

svc$operation(...),

where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- apigateway(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
```
region = "string"
)}

Operations

create_api_key
create_authorizer
create_base_path_mapping
create_deployment
create_documentation_part
create_documentation_version
create_domain_name
create_model
create_request_validator
create_resource
create_rest_api
create_stage
create_usage_plan
create_usage_plan_key
create_vpc_link
delete_api_key
delete_authorizer
delete_base_path_mapping
delete_client_certificate
delete_deployment
delete_documentation_part
delete_documentation_version
delete_domain_name
delete_gateway_response
delete_integration
delete_integration_response
delete_method
delete_method_response
delete_model
delete_request_validator
delete_resource
delete_rest_api
delete_stage
delete_usage_plan
delete_usage_plan_key
delete_vpc_link
flush_stage_authorizers_cache
flush_stage_cache
generate_client_certificate
get_account
get_api_key
get_api_keys
get_authorizer

Create an ApiKey resource
Adds a new Authorizer resource to an existing RestApi resource
Creates a new BasePathMapping resource
Creates a Deployment resource, which makes a specified RestApi callable over the internet
Creates a documentation part
Creates a documentation version
Creates a new domain name
Adds a new Model resource to an existing RestApi resource
Creates a RequestValidator of a given RestApi
Creates a Resource resource
Creates a new RestApi resource
Creates a new Stage resource that references a pre-existing Deployment for the API
Creates a usage plan with the throttle and quota limits, as well as the associated API stages,
Creates a usage plan key for adding an existing API key to a usage plan
Creates a VPC link, under the caller’s account in a selected region, in an asynchronous operation
Deletes the ApiKey resource
Deletes an existing Authorizer resource
Deletes the BasePathMapping resource
Deletes the ClientCertificate resource
Deletes a Deployment resource
Deletes a documentation part
Deletes a documentation version
Deletes the DomainName resource
Clears any customization of a GatewayResponse of a specified response type on the given RestApi
Represents a delete integration
Represents a delete integration response
Deletes an existing Method resource
Deletes an existing MethodResponse resource
Deletes a model
Deletes a RequestValidator of a given RestApi
Deletes a Resource resource
Deletes the specified API
Deletes a Stage resource
Deletes a usage plan of a given plan Id
Deletes a usage plan key and remove the underlying API key from the associated usage plan
Deletes an existing VpcLink of a specified identifier
Flushes all authorizer cache entries on a stage
Flushes a stage’s cache
Generates a ClientCertificate resource
Gets information about the current Account resource
Gets information about the current ApiKey resource
Gets information about the current ApiKeys resource
Describe an existing Authorizer resource
get_authorizers
get_base_path_mapping
get_base_path_mappings
get_client_certificate
get_client_certificates
get_deployment
get_deployments
get_documentation_part
get_documentation_parts
get_documentation_version
get_documentation_versions
get_domain_name
get_domain_names
get_export
get_gateway_response
get_gateway_responses
get_integration
get_integration_response
get_method
get_method_response
get_model
get_models
get_model_template
get_request_validator
get_request_validators
get_resource
get_resources
get_rest_api
get_rest_apis
get_sdk
get_sdk_type
get_sdk_types
get_stage
get_stages
get_tags
get_usage
get_usage_plan
get_usage_plan_key
get_usage_plan_keys
get_usage_plans
get_vpc_link
get_vpc_links
import_api_keys
import_documentation_parts
import_rest_api
put_gateway_response
put_integration
put_integration_response

Describe an existing Authorizers resource
Describe a BasePathMapping resource
Represents a collection of BasePathMapping resources
Gets information about the current ClientCertificate resource
Gets a collection of ClientCertificate resources
Gets information about a Deployment resource
Gets information about a Deployments collection
Gets a documentation part
Gets a documentation parts
Gets a documentation version
Gets documentation versions
Represents a domain name that is contained in a simpler, more intuitive URL that can be called
Represents a collection of DomainName resources
Exports a deployed version of a RestApi in a specified format
Gets a GatewayResponse of a specified response type on the given RestApi
Gets the GatewayResponses collection on the given RestApi
Get the integration settings
Represents a get integration response
Describe an existing Method resource
Describes a MethodResponse resource
Describes an existing model defined for a RestApi resource
Describes existing Models defined for a RestApi resource
Generates a sample mapping template that can be used to transform a payload into the structure of a model
Gets a RequestValidator of a given RestApi
Gets the RequestValidators collection of a given RestApi
Lists information about a resource
Lists information about a collection of Resource resources
Lists the RestApi resource in the collection
Lists the RestApis resources for your collection
Generates a client SDK for a RestApi and Stage
Gets an SDK type
Gets SDK types
Gets information about a Stage resource
Gets information about one or more Stage resources
Gets the Tags collection for a given resource
Gets the usage data of a usage plan in a specified time interval
Gets a usage plan of a given plan identifier
Gets a usage plan key of a given key identifier
Gets all the usage plan keys representing the API keys added to a specified usage plan
Gets all the usage plans of the caller’s account
Gets a specified VPC link under the caller’s account in a region
Gets the VpcLinks collection under the caller’s account in a selected region
Import API keys from an external source, such as a CSV-formatted file
Imports documentation parts
A feature of the API Gateway control service for creating a new API from an external API definition
Creates a customization of a GatewayResponse of a specified response type and status code
Sets up a method’s integration
Represents a put integration
<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>put_method</code></td>
<td>Add a method to an existing Resource resource</td>
</tr>
<tr>
<td><code>put_method_response</code></td>
<td>Adds a MethodResponse to an existing Method resource</td>
</tr>
<tr>
<td><code>put_rest_api</code></td>
<td>A feature of the API Gateway control service for updating an existing API with an input of external API definitions</td>
</tr>
<tr>
<td><code>tag_resource</code></td>
<td>Adds or updates a tag on a given resource</td>
</tr>
<tr>
<td><code>test_invoke_authorizer</code></td>
<td>Simulate the execution of an Authorizer in your RestApi with headers, parameters, and an incoming request body</td>
</tr>
<tr>
<td><code>test_invoke_method</code></td>
<td>Simulate the invocation of a Method in your RestApi with headers, parameters, and an incoming request body</td>
</tr>
<tr>
<td><code>untag_resource</code></td>
<td>Removes a tag from a given resource</td>
</tr>
<tr>
<td><code>update_account</code></td>
<td>Changes information about the current Account resource</td>
</tr>
<tr>
<td><code>update_api_key</code></td>
<td>Changes information about an ApiKey resource</td>
</tr>
<tr>
<td><code>update_authorizer</code></td>
<td>Updates an existing Authorizer resource</td>
</tr>
<tr>
<td><code>update_base_path_mapping</code></td>
<td>Changes information about the BasePathMapping resource</td>
</tr>
<tr>
<td><code>update_client_certificate</code></td>
<td>Changes information about an ClientCertificate resource</td>
</tr>
<tr>
<td><code>update_deployment</code></td>
<td>Changes information about a Deployment resource</td>
</tr>
<tr>
<td><code>update_documentation_part</code></td>
<td>Updates a documentation part</td>
</tr>
<tr>
<td><code>update_documentation_version</code></td>
<td>Updates a documentation version</td>
</tr>
<tr>
<td><code>update_domain_name</code></td>
<td>Changes information about the DomainName resource</td>
</tr>
<tr>
<td><code>update_gateway_response</code></td>
<td>Updates a GatewayResponse of a specified response type on the given RestApi</td>
</tr>
<tr>
<td><code>update_integration</code></td>
<td>Represents an update integration</td>
</tr>
<tr>
<td><code>update_integration_response</code></td>
<td>Represents an update integration response</td>
</tr>
<tr>
<td><code>update_method</code></td>
<td>Updates an existing Method resource</td>
</tr>
<tr>
<td><code>update_method_response</code></td>
<td>Updates an existing MethodResponse resource</td>
</tr>
<tr>
<td><code>update_model</code></td>
<td>Changes information about a model</td>
</tr>
<tr>
<td><code>update_request_validator</code></td>
<td>Updates a RequestValidator of a given RestApi</td>
</tr>
<tr>
<td><code>update_resource</code></td>
<td>Changes information about a Resource resource</td>
</tr>
<tr>
<td><code>update_rest_api</code></td>
<td>Changes information about the specified API</td>
</tr>
<tr>
<td><code>update_stage</code></td>
<td>Changes information about a Stage resource</td>
</tr>
<tr>
<td><code>update_usage</code></td>
<td>Grants a temporary extension to the remaining quota of a usage plan associated with a specified API key</td>
</tr>
<tr>
<td><code>update_usage_plan</code></td>
<td>Updates a usage plan of a given plan Id</td>
</tr>
<tr>
<td><code>update_vpc_link</code></td>
<td>Updates an existing VpcLink of a specified identifier</td>
</tr>
</tbody>
</table>

### Examples

```r
## Not run:
svc <- apigateway()
svc$create_api_key(
  Foo = 123
)
## End(Not run)
```

---

AmazonApiGatewayManagementApi
**Description**

The Amazon API Gateway Management API allows you to directly manage runtime aspects of your deployed APIs. To use it, you must explicitly set the SDK’s endpoint to point to the endpoint of your deployed API. The endpoint will be of the form https://{api-id}.execute-api.{region}.amazonaws.com/{stage}, or will be the endpoint corresponding to your API’s custom domain and base path, if applicable.

**Usage**

```python
apigatewaymanagementapi(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

**Arguments**

- `config`: Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
    - `creds`:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
    - `anonymous`: Set anonymous credentials.
  - `endpoint`: The complete URL to use for the constructed client.
  - `region`: The AWS Region used in instantiating the client.
  - `close_connection`: Immediately close all HTTP connections.
  - `timeout`: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - `s3_force_path_style`: Set this to `true` to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  - `sts_regional_endpoint`: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html

- `credentials`: Optional credentials shorthand for the `config` parameter
  - `creds`:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - `profile`: The name of a profile to use. If not given, then the default profile is used.
  - `anonymous`: Set anonymous credentials.

- `endpoint`: Optional shorthand for complete URL to use for the constructed client.

- `region`: Optional shorthand for AWS Region used in instantiating the client.
Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- apigatewaymanagementapi(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `delete_connection`  Delete the connection with the provided id
- `get_connection`    Get information about the connection with the provided id
- `post_to_connection` Sends the provided data to the specified connection
Examples

```r
## Not run:
svc <- apigatewaymanagementapi()
svc$delete_connection(
   Foo = 123
)
## End(Not run)
```

Description

Amazon API Gateway V2

Usage

```r
apigatewayv2(
   config = list(),
   credentials = list(),
   endpoint = NULL,
   region = NULL
)
```

Arguments

<table>
<thead>
<tr>
<th>config</th>
</tr>
</thead>
</table>
| Optional configuration of credentials, endpoint, and/or region.

- **credentials:**
  - **creds:**
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - `profile`: The name of a profile to use. If not given, then the default profile is used.
  - `anonymous`: Set anonymous credentials.
- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials**  
Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.

- **anonymous**: Set anonymous credentials.

**endpoint**  
Optional shorthand for complete URL to use for the constructed client.

**region**  
Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- apigatewayv2(
    config = list(
        credentials = list(
            creds = list(
                access_key_id = "string",
                secret_access_key = "string",
                session_token = "string"
            ),
            profile = "string",
            anonymous = "logical"
        ),
        endpoint = "string",
        region = "string",
        close_connection = "logical",
        timeout = "numeric",
        s3_force_path_style = "logical",
        sts_regional_endpoint = "string"
    ),
    credentials = list(
        creds = list(
            access_key_id = "string",
            secret_access_key = "string",
            session_token = "string"
        ),
        profile = "string",
```
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

create_api
create_api_mapping
create_authorizer
create_deployment
create_domain_name
create_integration
create_integration_response
create_model
create_route
create_route_response
create_stage
create_vpc_link
delete_access_log_settings
delete_api
delete_api_mapping
delete_authorizer
delete_cors_configuration
delete_deployment
delete_domain_name
delete_integration
delete_integration_response
delete_model
delete_route
delete_route_request_parameter
delete_route_response
delete_route_settings
delete_stage
delete_vpc_link
export_api
get_api
get_api_mapping
get_api_mappings
get_apis
get_authorizer
get_authorizers
get_deployment
get_deployments
get_domain_name
get_domain_names
get_integration

Creates an Api resource
Creates an API mapping
Creates an Authorizer for an API
Creates a Deployment for an API
Creates a domain name
Creates an Integration
Creates an IntegrationResponses
Creates a Model for an API
Creates a Route for an API
Creates a RouteResponse for a Route
Creates a Stage for an API
Creates a VPC link
Deletes the AccessLogSettings for a Stage
Deletes an Api resource
Deletes an API mapping
Deletes an Authorizer
Deletes a CORS configuration
Deletes a Deployment
Deletes a domain name
Deletes an Integration
Deletes an IntegrationResponses
Deletes a Model
Deletes a Route
Deletes a route request parameter
Deletes a RouteResponse
Deletes the RouteSettings for a stage
Deletes a Stage
Deletes a VPC link
Export api
Gets an Api resource
Gets an API mapping
Gets API mappings
Gets a collection of Api resources
Gets an Authorizer
Gets the Authorizers for an API
Gets a Deployment
Gets the Deployments for an API
Gets a domain name
Gets the domain names for an AWS account
Gets an Integration
<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>get_integration_response</td>
<td>Gets an IntegrationResponses</td>
</tr>
<tr>
<td>get_integration_responses</td>
<td>Gets the IntegrationResponses for an Integration</td>
</tr>
<tr>
<td>get_integrations</td>
<td>Gets the Integrations for an API</td>
</tr>
<tr>
<td>get_model</td>
<td>Gets a Model</td>
</tr>
<tr>
<td>get_models</td>
<td>Gets the Models for an API</td>
</tr>
<tr>
<td>get_model_template</td>
<td>Gets a model template</td>
</tr>
<tr>
<td>get_route</td>
<td>Gets a Route</td>
</tr>
<tr>
<td>get_route_response</td>
<td>Gets a RouteResponse</td>
</tr>
<tr>
<td>get_route_responses</td>
<td>Gets the RouteResponses for a Route</td>
</tr>
<tr>
<td>get_routes</td>
<td>Gets the Routes for an API</td>
</tr>
<tr>
<td>get_stage</td>
<td>Gets a Stage</td>
</tr>
<tr>
<td>get_stages</td>
<td>Gets the Stages for an API</td>
</tr>
<tr>
<td>get_tags</td>
<td>Gets a collection of Tag resources</td>
</tr>
<tr>
<td>get_vpc_link</td>
<td>Gets a VPC link</td>
</tr>
<tr>
<td>get_vpc_links</td>
<td>Gets a collection of VPC links</td>
</tr>
<tr>
<td>import_api</td>
<td>Imports an API</td>
</tr>
<tr>
<td>reimport_api</td>
<td>Puts an Api resource</td>
</tr>
<tr>
<td>reset_authorizers_cache</td>
<td>Resets all authorizer cache entries on a stage</td>
</tr>
<tr>
<td>tag_resource</td>
<td>Creates a new Tag resource to represent a tag</td>
</tr>
<tr>
<td>untag_resource</td>
<td>Deletes a Tag</td>
</tr>
<tr>
<td>update_api</td>
<td>Updates an Api resource</td>
</tr>
<tr>
<td>update_api_mapping</td>
<td>The API mapping</td>
</tr>
<tr>
<td>update_authorizer</td>
<td>Updates an Authorizer</td>
</tr>
<tr>
<td>update_deployment</td>
<td>Updates a Deployment</td>
</tr>
<tr>
<td>update_domain_name</td>
<td>Updates a domain name</td>
</tr>
<tr>
<td>update_integration</td>
<td>Updates an Integration</td>
</tr>
<tr>
<td>update_integration_response</td>
<td>Updates an IntegrationResponses</td>
</tr>
<tr>
<td>update_model</td>
<td>Updates a Model</td>
</tr>
<tr>
<td>update_route</td>
<td>Updates a Route</td>
</tr>
<tr>
<td>update_route_response</td>
<td>Updates a RouteResponse</td>
</tr>
<tr>
<td>update_stage</td>
<td>Updates a Stage</td>
</tr>
<tr>
<td>update_vpc_link</td>
<td>Updates a VPC link</td>
</tr>
</tbody>
</table>

### Examples

```r
## Not run:
svc <- apigatewayv2()
svc$create_api(
  Foo = 123
)
```

## End(Not run)
Description

Amazon Web Services AppFabric quickly connects software as a service (SaaS) applications across your organization. This allows IT and security teams to easily manage and secure applications using a standard schema, and employees can complete everyday tasks faster using generative artificial intelligence (AI). You can use these APIs to complete AppFabric tasks, such as setting up audit log ingestions or viewing user access. For more information about AppFabric, including the required permissions to use the service, see the Amazon Web Services AppFabric Administration Guide. For more information about using the Command Line Interface (CLI) to manage your AppFabric resources, see the AppFabric section of the CLI Reference.

Usage

```python
appfabric(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**: 
    - **creds**:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

- **credentials**: Optional credentials shorthand for the config parameter
• **creds:**
  - **access_key_id:** AWS access key ID
  - **secret_access_key:** AWS secret access key
  - **session_token:** AWS temporary session token

• **profile:** The name of a profile to use. If not given, then the default profile is used.

• **anonymous:** Set anonymous credentials.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- appfabric(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```
Operations

**batch_get_user_access_tasks**  Gets user access details in a batch request
**connect_app_authorization**  Establishes a connection between Amazon Web Services AppFabric and an application, which allows AppFabric to call the APIs of the application
**create_app_authorization**  Creates an app authorization within an app bundle, which allows AppFabric to connect to an application
**create_app_bundle**  Creates an app bundle to collect data from an application using AppFabric
**create_ingestion**  Creates an ingestion for an application
**create_ingestion_destination**  Creates an ingestion destination, which specifies how an application’s ingested data is processed and where it’s delivered
**delete_app_authorization**  Deletes an app authorization
**delete_app_bundle**  Deletes an app bundle
**delete_ingestion**  Deletes an ingestion
**delete_ingestion_destination**  Deletes an ingestion destination
**get_app_authorization**  Returns information about an app authorization
**get_app_bundle**  Returns information about an app bundle
**get_ingestion**  Returns information about an ingestion
**get_ingestion_destination**  Returns information about an ingestion destination
**list_app_authorizations**  Returns a list of all app authorizations configured for an app bundle
**list_app_bundles**  Returns a list of app bundles
**list_ingestion_destinations**  Returns a list of all ingestion destinations configured for an ingestion
**list_ingestions**  Returns a list of all ingestions configured for an app bundle
**list_tags_for_resource**  Returns a list of tags for a resource
**start_ingestion**  Starts (enables) an ingestion, which collects data from an application
**start_user_access_tasks**  Starts the tasks to search user access status for a specific email address
**stop_ingestion**  Stops (disables) an ingestion
**tag_resource**  Assigns one or more tags (key-value pairs) to the specified resource
**untag_resource**  Removes a tag or tags from a resource
**update_app_authorization**  Updates an app authorization within an app bundle, which allows AppFabric to connect to an application
**update_ingestion_destination**  Updates an ingestion destination, which specifies how an application’s ingested data is processed and where it’s delivered

Examples

```r
## Not run:
svc <- appfabric()
svc$batch_get_user_access_tasks(
   Foo = 123
)

## End(Not run)
```
Description

With Application Auto Scaling, you can configure automatic scaling for the following resources:

- Amazon AppStream 2.0 fleets
- Amazon Aurora Replicas
- Amazon Comprehend document classification and entity recognizer endpoints
- Amazon DynamoDB tables and global secondary indexes throughput capacity
- Amazon ECS services
- Amazon ElastiCache for Redis clusters (replication groups)
- Amazon EMR clusters
- Amazon Keypaces (for Apache Cassandra) tables
- Lambda function provisioned concurrency
- Amazon Managed Streaming for Apache Kafka broker storage
- Amazon Neptune clusters
- Amazon SageMaker endpoint variants
- Amazon SageMaker Serverless endpoint provisioned concurrency
- Amazon SageMaker inference components
- Spot Fleets (Amazon EC2)
- Custom resources provided by your own applications or services

To learn more about Application Auto Scaling, see the Application Auto Scaling User Guide.

API Summary

The Application Auto Scaling service API includes three key sets of actions:

- Register and manage scalable targets - Register Amazon Web Services or custom resources as scalable targets (a resource that Application Auto Scaling can scale), set minimum and maximum capacity limits, and retrieve information on existing scalable targets.
- Configure and manage automatic scaling - Define scaling policies to dynamically scale your resources in response to CloudWatch alarms, schedule one-time or recurring scaling actions, and retrieve your recent scaling activity history.
- Suspend and resume scaling - Temporarily suspend and later resume automatic scaling by calling the register_scalable_target API action for any Application Auto Scaling scalable target. You can suspend and resume (individually or in combination) scale-out activities that are triggered by a scaling policy, scale-in activities that are triggered by a scaling policy, and scheduled scaling.

Usage

```python
applicationautoscaling(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```
Arguments

Optional configuration of credentials, endpoint, and/or region.

- **credentials**:
  - **creds**:
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
    - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.
- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

Credentials

Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

Endpoint

Optional shorthand for complete URL to use for the constructed client.

Region

Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- applicationautoscaling(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string",
        profile = "string",
        anonymous = TRUE
      )
    ),
    endpoint = "string",
    region = "string",
    close_connection = TRUE,
    timeout = 60,
    s3_force_path_style = TRUE,
    sts_regional_endpoint = "string"
  )
)
```
secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
),
    credentials = list(
        creds = list(
            access_key_id = "string",
            secret_access_key = "string",
            session_token = "string"
        ),
        profile = "string",
        anonymous = "logical"
    ),
    endpoint = "string",
    region = "string"
)

Operations

delete_scaling_policy  Deletes the specified scaling policy for an Application Auto Scaling scalable target
delete_scheduled_action  Deletes the specified scheduled action for an Application Auto Scaling scalable target
deregister_scalable_target  Deregisters an Application Auto Scaling scalable target when you have finished using it
describe_scalable_targets  Provides descriptive information about the scalable targets in the specified namespace
describe_scaling_activities  Describes the Application Auto Scaling scaling activities in the specified namespace from the previous six weeks
describe_scaling_policies  Describes the Application Auto Scaling scaling policies for the specified service namespace
describe_scheduled_actions  Describes the Application Auto Scaling scheduled actions for the specified service namespace
describe_scalable_targets  Gets information about the scalable targets
list_tags_for_resource  Returns all the tags on the specified Application Auto Scaling scalable target
put_scaling_policy  Creates or updates a scaling policy for an Application Auto Scaling scalable target
put_scheduled_action  Creates or updates a scheduled action for an Application Auto Scaling scalable target
register_scalable_target  Registers or updates a scalable target, which is the resource that you want to scale
tag_resource  Adds or edits tags on an Application Auto Scaling scalable target
untag_resource  Deletes tags from an Application Auto Scaling scalable target

Examples

## Not run:
svc <- applicationautoscaling()
This example deletes a scaling policy for the Amazon ECS service called
web-app, which is running in the default cluster.

svc$delete_scaling_policy(
    PolicyName = "web-app-cpu-lt-25",
    ResourceId = "service/default/web-app",
    ScalableDimension = "ecs:service:DesiredCount",
    ServiceNamespace = "ecs"
)

## End(Not run)

---

AWS Application Cost Profiler

Description

This reference provides descriptions of the AWS Application Cost Profiler API.
The AWS Application Cost Profiler API provides programmatic access to view, create, update, and delete application cost report definitions, as well as to import your usage data into the Application Cost Profiler service.

For more information about using this service, see the AWS Application Cost Profiler User Guide.

Usage

applicationcostprofiler(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

- config: Optional configuration of credentials, endpoint, and/or region.
  - credentials:
    - creds:
      - access_key_id: AWS access key ID
      - secret_access_key: AWS secret access key
      - session_token: AWS temporary session token
    - profile: The name of a profile to use. If not given, then the default profile is used.
    - anonymous: Set anonymous credentials.
  - endpoint: The complete URL to use for the constructed client.
  - region: The AWS Region used in instantiating the client.
• close_connection: Immediately close all HTTP connections.
• timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.
region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

svc <- applicationcostprofiler(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(}
creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

delete_report_definition  Deletes the specified report definition in AWS Application Cost Profiler
get_report_definition     Retrieves the definition of a report already configured in AWS Application Cost Profiler
import_application_usage Ingests application usage data from Amazon Simple Storage Service (Amazon S3)
list_report_definitions   Retrieves a list of all reports and their configurations for your AWS account
put_report_definition     Creates the report definition for a report in Application Cost Profiler
update_report_definition  Updates existing report in AWS Application Cost Profiler

Examples

```r
## Not run:
svc <- applicationcostprofiler()
svc$delete_report_definition(
    Foo = 123
)

## End(Not run)
```

Amazon CloudWatch Application Insights

Amazon CloudWatch Application Insights is a service that helps you detect common problems with your applications. It enables you to pinpoint the source of issues in your applications (built with technologies such as Microsoft IIS, .NET, and Microsoft SQL Server), by providing key insights into detected problems.

After you onboard your application, CloudWatch Application Insights identifies, recommends, and sets up metrics and logs. It continuously analyzes and correlates your metrics and logs for unusual behavior to surface actionable problems with your application. For example, if your application is slow and unresponsive and leading to HTTP 500 errors in your Application Load Balancer (ALB),
Application Insights informs you that a memory pressure problem with your SQL Server database is occurring. It bases this analysis on impactful metrics and log errors.

Usage

```python
applicationinsights(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**: Optional credentials shorthand for the config parameter
    - **creds**:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

- **endpoint**: Optional shorthand for complete URL to use for the constructed client.
- **region**: Optional shorthand for AWS Region used in instantiating the client.
Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- applicationinsights(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

<table>
<thead>
<tr>
<th>Operation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>add_workload</td>
<td>Adds a workload to a component</td>
</tr>
<tr>
<td>create_application</td>
<td>Adds an application that is created from a resource group</td>
</tr>
<tr>
<td>create_component</td>
<td>Creates a custom component by grouping similar standalone instances</td>
</tr>
<tr>
<td>create_log_pattern</td>
<td>Adds a log pattern to a LogPatternSet</td>
</tr>
<tr>
<td>delete_application</td>
<td>Removes the specified application from monitoring</td>
</tr>
<tr>
<td>delete_component</td>
<td>Ungroups a custom component</td>
</tr>
<tr>
<td>delete_log_pattern</td>
<td>Removes the specified log pattern from a LogPatternSet</td>
</tr>
<tr>
<td>describe_application</td>
<td>Describes the application</td>
</tr>
</tbody>
</table>
describe_component
describe_component_configuration
describe_component_configuration_recommendation
describe_log_pattern
describe_observation
describe_problem
describe_problem_observations
describe_workload
list_applications
list_components
list_configuration_history
list_log_patterns
list_log_pattern_sets
list_problems
list_tags_for_resource
list_workloads
remove_workload
tag_resource
untag_resource
update_application
update_component
update_component_configuration
update_log_pattern
update_problem
update_workload

Describes a component and lists the resources that are grouped together
Describes the monitoring configuration of the component
Describes the recommended monitoring configuration of the component
Describe a specific log pattern from a LogPatternSet
Describes an anomaly or error with the application
Describes an application problem
Describes the anomalies or errors associated with the problem
Describes a workload and its configuration
Lists the IDs of the applications that you are monitoring
Lists the auto-grouped, standalone, and custom components of the application
Lists the INFO, WARN, and ERROR events for periodic configuration updates performed by Application Insights
Lists the log patterns in the specific log LogPatternSet
Lists the log pattern sets in the specific application
Lists the problems with your application
Retrieve a list of the tags (keys and values) that are associated with a specific resource
Lists the workloads that are configured on a given component
Remove workload from a component
Add one or more tags (keys and values) to a specified application
Remove one or more tags (keys and values) from a specified application
Updates the application
Updates the custom component name and/or the list of resources that make up the component
Updates the monitoring configurations for the component
Adds a log pattern to a LogPatternSet
Updates the visibility of the problem or specifies the problem as RESOLVED
Adds a workload to a component

Examples

```r
## Not run:
svc <- applicationinsights()
svc$add_workload(
   Foo = 123
)

## End(Not run)
```

appmesh

AWS App Mesh

Description

App Mesh is a service mesh based on the Envoy proxy that makes it easy to monitor and control microservices. App Mesh standardizes how your microservices communicate, giving you end-to-end visibility and helping to ensure high availability for your applications.
App Mesh gives you consistent visibility and network traffic controls for every microservice in an application. You can use App Mesh with Amazon Web Services Fargate, Amazon ECS, Amazon EKS, Kubernetes on Amazon Web Services, and Amazon EC2.

App Mesh supports microservice applications that use service discovery naming for their components. For more information about service discovery on Amazon ECS, see Service Discovery in the Amazon Elastic Container Service Developer Guide. Kubernetes kube-dns and coredns are supported. For more information, see DNS for Services and Pods in the Kubernetes documentation.

Usage

appmesh(config = list(), credentials = list(), endpoint = NULL, region = NULL)

Arguments

config  Optional configuration of credentials, endpoint, and/or region.

- credentials:
  - creds:
    - access_key_id: AWS access key ID
    - secret_access_key: AWS secret access key
    - session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.

- endpoint: The complete URL to use for the constructed client.

- region: The AWS Region used in instantiating the client.

- close_connection: Immediately close all HTTP connections.

- timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

- s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

- sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials  Optional credentials shorthand for the config parameter

- creds:
  - access_key_id: AWS access key ID
  - secret_access_key: AWS secret access key
  - session_token: AWS temporary session token

- profile: The name of a profile to use. If not given, then the default profile is used.

- anonymous: Set anonymous credentials.

endpoint  Optional shorthand for complete URL to use for the constructed client.

region  Optional shorthand for AWS Region used in instantiating the client.
Value

A client for the service. You can call the service's operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- appmesh(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `create_gateway_route` Creates a gateway route
- `create_mesh` Creates a service mesh
- `create_route` Creates a route that is associated with a virtual router
- `create_virtual_gateway` Creates a virtual gateway
- `create_virtual_node` Creates a virtual node within a service mesh
- `create_virtual_router` Creates a virtual router within a service mesh
- `create_virtual_service` Creates a virtual service within a service mesh
- `delete_gateway_route` Deletes an existing gateway route
delete_mesh Deletes an existing service mesh
delete_route Deletes an existing route
delete_virtual_gateway Deletes an existing virtual gateway
delete_virtual_node Deletes an existing virtual node
delete_virtual_router Deletes an existing virtual router
delete_virtual_service Deletes an existing virtual service
describe_gateway_route Describes an existing gateway route
describe_mesh Describes an existing service mesh
describe_route Describes an existing route
describe_virtual_gateway Describes an existing virtual gateway
describe_virtual_node Describes an existing virtual node
describe_virtual_router Describes an existing virtual router
describe_virtual_service Describes an existing virtual service
list_gateway_routes Returns a list of existing gateway routes that are associated to a virtual gateway
list_meshes Returns a list of existing service meshes
list_routes Returns a list of existing routes in a service mesh
list_tags_for_resource List the tags for an App Mesh resource
list_virtual_gateways Returns a list of existing virtual gateways in a service mesh
list_virtual_nodes Returns a list of existing virtual nodes
list_virtual_routers Returns a list of existing virtual routers in a service mesh
list_virtual_services Returns a list of existing virtual services in a service mesh
tag_resource Associates the specified tags to a resource with the specified resourceArn
untag_resource Deletes specified tags from a resource
update_gateway_route Updates an existing gateway route that is associated to a specified virtual gateway in a service mesh
update_mesh Updates an existing service mesh
update_route Updates an existing route for a specified service mesh and virtual router
update_virtual_gateway Updates an existing virtual gateway in a specified service mesh
update_virtual_node Updates an existing virtual node in a specified service mesh
update_virtual_router Updates an existing virtual router in a specified service mesh
update_virtual_service Updates an existing virtual service in a specified service mesh

Examples

```r
## Not run:
svc <- appmesh()
svc$create_gateway_route(
  Foo = 123
)

## End(Not run)
```

appregistry AWS Service Catalog App Registry
Description

Amazon Web Services Service Catalog AppRegistry enables organizations to understand the application context of their Amazon Web Services resources. AppRegistry provides a repository of your applications, their resources, and the application metadata that you use within your enterprise.

Usage

```python
appregistry(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
      - **anonymous**: Set anonymous credentials.
    - **endpoint**: The complete URL to use for the constructed client.
    - **region**: The AWS Region used in instantiating the client.
    - **close_connection**: Immediately close all HTTP connections.
    - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
    - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
    - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.

- **endpoint**: Optional shorthand for complete URL to use for the constructed client.

- **region**: Optional shorthand for AWS Region used in instantiating the client.
Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- appregistry(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- **associate_attribute_group**: Associates an attribute group with an application to augment the application’s metadata
- **associate_resource**: Associates a resource with an application
- **create_application**: Creates a new application that is the top-level node in a hierarchy of related cloud resource abstractions
- **create_attribute_group**: Creates a new attribute group as a container for user-defined attributes
- **delete_application**: Deletes an application that is specified either by its application ID, name, or ARN
- **delete_attribute_group**: Deletes an attribute group, specified either by its attribute group ID, name, or ARN
- **disassociate_attribute_group**: Disassociates an attribute group from an application to remove the extra attributes contained in the group
- **disassociate_resource**: Disassociates a resource from application
get_application  Retrieves metadata information about one of your applications
get_associated_resource  Gets the resource associated with the application
get_attribute_group  Retrieves an attribute group by its ARN, ID, or name
get_configuration  Retrieves a TagKey configuration from an account
list_applications  Retrieves a list of all of your applications
list_associated_attribute_groups  Lists all attribute groups that are associated with specified application
list_associated_resources  Lists all of the resources that are associated with the specified application
list_attribute_groups  Lists all attribute groups which you have access to
list_attribute_groups_for_application  Lists the details of all attribute groups associated with a specific application
list_tags_for_resource  Lists all of the tags on the resource
put_configuration  Associates a TagKey configuration to an account
sync_resource  Syncs the resource with current AppRegistry records
tag_resource  Assigns one or more tags (key-value pairs) to the specified resource
untag_resource  Removes tags from a resource
update_application  Updates an existing application with new attributes
update_attribute_group  Updates an existing attribute group with new details

Examples

```r
## Not run:
svc <- appregistry()
svc$associate_attribute_group(
  Foo = 123
)
## End(Not run)
```

**apprunner**  
*AWS App Runner*

---

**Description**

App Runner

App Runner is an application service that provides a fast, simple, and cost-effective way to go directly from an existing container image or source code to a running service in the Amazon Web Services Cloud in seconds. You don’t need to learn new technologies, decide which compute service to use, or understand how to provision and configure Amazon Web Services resources.

App Runner connects directly to your container registry or source code repository. It provides an automatic delivery pipeline with fully managed operations, high performance, scalability, and security.

For more information about App Runner, see the [App Runner Developer Guide](#). For release information, see the [App Runner Release Notes](#).
To install the Software Development Kits (SDKs), Integrated Development Environment (IDE) Toolkits, and command line tools that you can use to access the API, see Tools for Amazon Web Services.

Endpoints

For a list of Region-specific endpoints that App Runner supports, see App Runner endpoints and quotas in the Amazon Web Services General Reference.

Usage

```r
apprunner(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

Arguments

- `config` Optional configuration of credentials, endpoint, and/or region.
  - credentials:
    - creds:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - profile: The name of a profile to use. If not given, then the default profile is used.
    - anonymous: Set anonymous credentials.
  - endpoint: The complete URL to use for the constructed client.
  - region: The AWS Region used in instantiating the client.
  - close_connection: Immediately close all HTTP connections.
  - timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  - sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends html

- `credentials` Optional credentials shorthand for the config parameter
  - creds:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.
endpoint  Optional shorthand for complete URL to use for the constructed client.
region    Optional shorthand for AWS Region used in instantiating the client.

Value
A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

svc <- apprunner(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)

Operations

associate_custom_domain  Associate your own domain name with the App Runner subdomain URL of your
                         App Runner service
create_auto_scaling_configuration  Create an App Runner automatic scaling configuration resource
create_connection  Create an App Runner connection resource
create_observability_configuration  Create an App Runner observability configuration resource
create_service
create_vpc_connector
create_vpc_ingress_connection
delete_auto_scaling_configuration
delete_connection
delete_observability_configuration
delete_service
delete_vpc_connector
delete_vpc_ingress_connection
describe_auto_scaling_configuration
describe_custom_domains
describe_observability_configuration
describe_service
describe_vpc_connector
describe_vpc_ingress_connection
disable_custom_domain
list_auto_scaling_configurations
list_connections
list_observability_configurations
list_operations
list_services
list_services_for_auto_scaling_configuration
list_tags_for_resource
list_vpc_connectors
list_vpc_ingress_connections
pause_service
resume_service
start_deployment
tag_resource
untag_resource
update_default_auto_scaling_configuration
update_service
update_vpc_ingress_connection

Create an App Runner service
Create an App Runner VPC connector resource
Create an App Runner VPC Ingress Connection resource
Delete an App Runner automatic scaling configuration resource
Delete an App Runner connection
Delete an App Runner observability configuration resource
Delete an App Runner service
Delete an App Runner VPC connector resource
Delete an App Runner VPC Ingress Connection resource that’s associated with an App Runner service
Return a full description of an App Runner automatic scaling configuration resource
Return a description of custom domain names that are associated with an App Runner service
Return a full description of an App Runner observability configuration resource
Return a full description of an App Runner service
Return a description of an App Runner VPC connector resource
Return a full description of an App Runner VPC Ingress Connection resource
Disassociate a custom domain name from an App Runner service
Returns a list of active App Runner automatic scaling configurations in your Amazon Web Services account
Returns a list of App Runner connections that are associated with your Amazon Web Services account
Returns a list of active App Runner observability configurations in your Amazon Web Services account
Return a list of operations that occurred on an App Runner service
Returns a list of running App Runner services in your Amazon Web Services account
Returns a list of the associated App Runner services using an auto scaling configuration
List tags that are associated with an App Runner resource
Returns a list of App Runner VPC connectors in your Amazon Web Services account
Return a list of App Runner VPC Ingress Connections in your Amazon Web Services account
Pause an active App Runner service
Resume an active App Runner service
Initiate a manual deployment of the latest commit in a source code repository or the latest image in a source image repository to an App Runner service
Add tags to, or update the tag values of, an App Runner resource
Remove tags from an App Runner resource
Update an auto scaling configuration to be the default
Update an App Runner service
Update an existing App Runner VPC Ingress Connection resource

Examples

## Not run:
svc <- apprunner()
svc$associate_custom_domain(
  Foo = 123
)

## End(Not run)
Description

Amazon AppStream 2.0

This is the *Amazon AppStream 2.0 API Reference*. This documentation provides descriptions and syntax for each of the actions and data types in AppStream 2.0. AppStream 2.0 is a fully managed, secure application streaming service that lets you stream desktop applications to users without rewriting applications. AppStream 2.0 manages the AWS resources that are required to host and run your applications, scales automatically, and provides access to your users on demand.

You can call the AppStream 2.0 API operations by using an interface VPC endpoint (interface endpoint). For more information, see Access AppStream 2.0 API Operations and CLI Commands Through an Interface VPC Endpoint in the *Amazon AppStream 2.0 Administration Guide*.

To learn more about AppStream 2.0, see the following resources:

- Amazon AppStream 2.0 product page
- Amazon AppStream 2.0 documentation

Usage

```r
appstream(config = list(),
          credentials = list(),
          endpoint = NULL,
          region = NULL)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.

• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

credentials Optional credentials shorthand for the config parameter

• **creds**:
  
  – **access_key_id**: AWS access key ID
  
  – **secret_access_key**: AWS secret access key
  
  – **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- appstream(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    )
  )
)
```
appstream

),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

associate_app_block_builder_app_block
associate_application_fleet
associate_application_to_entitlement
associate_fleet
batch_associate_user_stack
batch_disassociate_user_stack
copy_image
create_app_block
create_app_block_builder
create_app_block_builder_streaming_url
create_application
create_directory_config
create_entitlement
create_fleet
create_image_builder
create_image_builder_streaming_url
create_stack
create_streaming_url
create_updated_image
create_usage_report_subscription
create_user
delete_app_block
delete_app_block_builder
delete_application
delete_directory_config
delete_entitlement
delete_fleet
delete_image
delete_image_builder
delete_image_permissions
delete_stack
delete_usage_report_subscription
delete_user
describe_app_block_builder_app_block_associations
describe_app_block_builders
describe_app_blocks
describe_application_fleet_associations
describe_applications

Associates the specified app block builder with the specified app block
Associates the specified application with the specified fleet
Associates an application to entitle
Associates the specified fleet with the specified stack
Associates the specified users with the specified stacks
Disassociates the specified users from the specified stacks
Copies the image within the same region or to a new region within the
Create an app block
Create an app block builder
Create a URL to start a create app block builder streaming session
Create an application
Create a Directory Config object in AppStream 2
Create a new entitlement
Create a fleet
Create an image builder
Create a URL to start an image builder streaming session
Create a stack to start streaming applications to users
Create a temporary URL to start an AppStream 2
Create a new image with the latest Windows operating system updates
Create a usage report subscription
Create a new user in the user pool
Delete an app block
Delete an app block builder
Delete an application
Delete the specified Directory Config object from AppStream 2
Delete the specified entitlement
Delete the specified fleet
Delete the specified image
Delete the specified image builder and releases the capacity
Delete permissions for the specified private image
Delete the specified stack
Disables usage report generation
Deletes a user from the user pool
Retrieves a list that describes one or more app block builder associations
Retrieves a list that describes one or more app block builders
Retrieves a list that describes one or more app blocks
Retrieves a list that describes one or more application fleet associations
Retrieves a list that describes one or more applications
describe_directory_configs
describe_entitlements
describe_fleets
describe_image_builders
describe_image_permissions
describe_images
describe_sessions
describe_stacks
describe_usage_report_subscriptions
describe_users
describe_user_stack_associations
disable_user
disassociate_app_block_builder_app_block
disassociate_application_fleet
disassociate_application_from_entitlement
disassociate_fleet
enable_user
expire_session
list_associated_fleets
list_associated_stacks
list_entitled_applications
list_tags_for_resource
start_app_block_builder
start_fleet
start_image_builder
stop_app_block_builder
stop_fleet
stop_image_builder
tag_resource
untag_resource
update_app_block_builder
update_application
update_directory_config
update_entitlement
update_fleet
update_image_permissions
update_stack

Retrieves a list that describes one or more specified Directory Config objects
Retrieves a list that describes one or more entitlements
Retrieves a list that describes one or more specified fleets, if the fleet names are provided
Retrieves a list that describes one or more specified image builders, if the image builder names are provided
Retrieves a list that describes the permissions for shared AWS accounts
Retrieves a list that describes one or more specified images, if the image names or image ARNs are provided
Retrieves a list that describes one or more specified stacks, if the stack names are provided
Retrieves a list that describes one or more usage report subscriptions
Retrieves a list that describes one or more specified users in the user pool
Retrieves a list that describes the UserStackAssociation objects
Disassociates a specified app block builder from a specified app block
Disassociates the specified application from the fleet
Deletes the specified application from the specified entitlement
Disassociates the specified fleet from the specified stack
Enables a user in the user pool
Immediately stops the specified streaming session
Retrieves the name of the fleet that is associated with the specified stack
Retrieves the name of the stack with which the specified fleet is associated
Retrieves a list of entitled applications
Retrieves a list of all tags for the specified AppStream 2
Starts an app block builder
Starts the specified fleet
Starts the specified image builder
Stops an app block builder
Stops the specified fleet
Stops the specified image builder
Adds or overwrites one or more tags for the specified AppStream 2
Disassociates one or more specified tags from the specified AppStream 2
Updates an app block builder
Updates the specified application
Updates the specified Directory Config object in AppStream 2
Updates the specified entitlement
Updates the specified fleet
Adds or updates permissions for the specified private image
Updates the specified fields for the specified stack

Examples

## Not run:
svc <- appstream()
svc$associate_app_block_builder_app_block(
  Foo = 123
)

## End(Not run)
Description

Welcome to the Zonal Shift API Reference Guide for Amazon Route 53 Application Recovery Controller (Route 53 ARC).

You can start a zonal shift to move traffic for a load balancer resource away from an Availability Zone to help your application recover quickly from an impairment in an Availability Zone. For example, you can recover your application from a developer’s bad code deployment or from an Amazon Web Services infrastructure failure in a single Availability Zone.

You can also configure zonal autoshift for a load balancer resource. Zonal autoshift is a capability in Route 53 ARC where Amazon Web Services shifts away application resource traffic from an Availability Zone, on your behalf, to help reduce your time to recovery during events. Amazon Web Services shifts away traffic for resources that are enabled for zonal autoshift whenever Amazon Web Services determines that there’s an issue in the Availability Zone that could potentially affect customers.

To ensure that zonal autoshift is safe for your application, you must also configure practice runs when you enable zonal autoshift for a resource. Practice runs start weekly zonal shifts for a resource, to shift traffic for the resource out of an Availability Zone. Practice runs make sure, on a regular basis, that you have enough capacity in all the Availability Zones in an Amazon Web Services Region for your application to continue to operate normally when traffic for a resource is shifted away from one Availability Zone.

You must prescale resource capacity in all Availability Zones in the Region where your application is deployed, before you configure practice runs or enable zonal autoshift for a resource. You should not rely on scaling on demand when an autoshift or practice run starts.

For more information about using zonal shift and zonal autoshift, see the Amazon Route 53 Application Recovery Controller Developer Guide.

Usage

```python
arczonalshift(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

Arguments

- `config` Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
    - `creds`:
* access_key_id: AWS access key ID
* secret_access_key: AWS secret access key
* session_token: AWS temporary session token

- profile: The name of a profile to use. If not given, then the default profile is used.
- anonymous: Set anonymous credentials.

• endpoint: The complete URL to use for the constructed client.
• region: The AWS Region used in instantiating the client.
• close_connection: Immediately close all HTTP connections.
• timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter
• creds:
  - access_key_id: AWS access key ID
  - secret_access_key: AWS secret access key
  - session_token: AWS temporary session token
• profile: The name of a profile to use. If not given, then the default profile is used.
• anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.
region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- arczonalshift(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"))
```
Operations

- **cancel_zonal_shift**
  Cancel a zonal shift in Amazon Route 53 Application Recovery Controller

- **create_practice_run_configuration**
  A practice run configuration for zonal autoshift is required when you enable zonal autoshift.

- **delete_practice_run_configuration**
  Deletes the practice run configuration for a resource.

- **get_managed_resource**
  Get information about a resource that’s been registered for zonal shifts with Amazon Route 53 Application Recovery Controller.

- **list_autoshifts**
  Returns the active autoshifts for a specified resource.

- **list_managed_resources**
  Lists all the resources in your Amazon Web Services account in this Amazon Web Services Region.

- **list_zonal_shifts**
  Lists all active and completed zonal shifts in Amazon Route 53 Application Recovery Controller.

- **start_zonal_shift**
  You start a zonal shift to temporarily move load balancer traffic away from an Availability Zone in an Amazon Web Services Region, which can be triggered by a developer’s bad code deployment or from an Amazon Web Services infrastructure failure in a single Availability Zone.

- **update_practice_run_configuration**
  Update a practice run configuration to change one or more of the following: add, change, or remove the blocking alarm; change the outcome alarm; or add, change, or remove blocking dates or time windows.

- **update_zonal_autoshift_configuration**
  You can update the zonal autoshift status for a resource, to enable or disable zonal autoshift.

- **update_zonal_shift**
  Update an active zonal shift in Amazon Route 53 Application Recovery Controller.

Examples

```r
## Not run:
svc <- arczonalshift()
svc$cancel_zonal_shift(
  Foo = 123
)
```

## End(Not run)
Description

Amazon Athena is an interactive query service that lets you use standard SQL to analyze data directly in Amazon S3. You can point Athena at your data in Amazon S3 and run ad-hoc queries and get results in seconds. Athena is serverless, so there is no infrastructure to set up or manage. You pay only for the queries you run. Athena scales automatically—executing queries in parallel—so results are fast, even with large datasets and complex queries. For more information, see What is Amazon Athena in the Amazon Athena User Guide.

If you connect to Athena using the JDBC driver, use version 1.1.0 of the driver or later with the Amazon Athena API. Earlier version drivers do not support the API. For more information and to download the driver, see Accessing Amazon Athena with JDBC.

Usage

```r
athena(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
– **session_token**: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.

- **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**
A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- athena(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string",
  close_connection = "logical",
  timeout = "numeric",
  s3_force_path_style = "logical",
  sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

**Operations**
batch_get_named_query
batch_get_prepared_statement
batch_get_query_execution
cancel_capacity_reservation
create_capacity_reservation
create_data_catalog
create_named_query
create_notebook
create_prepared_statement
create_presigned_notebook_url
create_work_group
delete_capacity_reservation
delete_data_catalog
delete_named_query
delete_notebook
delete_prepared_statement
delete_work_group
export_notebook
get_calculation_execution
get_calculation_execution_code
get_calculation_execution_status
get_capacity_assignment_configuration
get_capacity_reservation
get_database
get_data_catalog
get_named_query
get_notebook_metadata
get_prepared_statement
get_query_execution
get_query_results
get_query_runtime_statistics
get_session
get_session_status
get_table_metadata
get_work_group
import_notebook
list_application_dpu_sizes
list_calculation_executions
list_capacity_reservations
list_databases
list_data_catalogs
list_engine_versions
list_executors
list_named_queries
list_notebook_metadata
list_notebook_sessions
list_prepared_statements
list_query_executions

Returns the details of a single named query or a list of up to 50 queries, which you provide as an array of query ID strings.

Returns the details of a single prepared statement or a list of up to 256 prepared statements for the array of prepared statement names that you provide.

Returns the details of a single query execution or a list of up to 50 query executions, which you provide as an array of query execution ID strings.

Cancels the capacity reservation with the specified name.

Creates a capacity reservation with the specified name and number of requested data processing units.

Creates (registers) a data catalog with the specified name and properties.

Creates a named query in the specified workgroup.

Creates an empty ipynb file in the specified Apache Spark enabled workgroup.

Creates a prepared statement for use with SQL queries in Athena.

Gets an empty ipynb file in the specified Apache Spark enabled workgroup.

Creates a workgroup with the specified name.

Deletes a named query if you have access to the workgroup in which the query was saved.

Deletes a cancelled capacity reservation.

Deletes a data catalog.

Deletes the named query if you have access to the workgroup in which the query was saved.

Deletes the specified notebook.

Deletes the prepared statement with the specified name from the specified workgroup.

Deletes the workgroup with the specified name.

Exports the specified notebook and its metadata.

Describes a previously submitted calculation execution.

Retrieves the unencrypted code that was executed for the calculation.

Gets the status of a current calculation.

Gets the capacity assignment configuration for a capacity reservation, if one exists.

Returns information about the capacity reservation with the specified name.

Returns a database object for the specified database and data catalog.

Returns the specified data catalog.

Returns information about a single query.

Retrieves notebook metadata for the specified notebook ID.

Retrieves the prepared statement with the specified name from the specified workgroup.

Returns information about a single execution of a query if you have access to the workgroup in which the query was saved.

Streams the results of a single query execution specified by QueryExecutionId from Amazon S3.

Returns query execution runtime statistics related to a single execution of a query if you have access to the workgroup in which the query was saved.

Gets the full details of a previously created session, including the session status and configuration.

Gets the current status of a session.

Returns table metadata for the specified catalog, database, and table.

Returns information about the workgroup with the specified name.

Imports a single ipynb file to a Spark enabled workgroup.

Returns the supported DPU sizes for the supported application runtimes (for example, Athena notebook version 1).

Lists the calculations that have been submitted to a session in descending order.

Lists the capacity reservations for the current account.

Lists the databases in the specified data catalog.

Lists the data catalogs in the current Amazon Web Services account.

Returns a list of engine versions that are available to choose from, including the Auto option.

Lists, in descending order, the executors that joined a session.

Provides a list of available query IDs only for queries saved in the specified workgroup.

Displays the notebook files for the specified workgroup in paginated format.

Lists, in descending order, the sessions that have been created in a notebook that are active.

Lists the prepared statements in the specified workgroup.

Provides a list of available query execution IDs for the queries in the specified workgroup.
Welcome to the Audit Manager API reference. This guide is for developers who need detailed information about the Audit Manager API operations, data types, and errors.

Audit Manager is a service that provides automated evidence collection so that you can continually audit your Amazon Web Services usage. You can use it to assess the effectiveness of your controls, manage risk, and simplify compliance.

Audit Manager provides prebuilt frameworks that structure and automate assessments for a given compliance standard. Frameworks include a prebuilt collection of controls with descriptions and
testing procedures. These controls are grouped according to the requirements of the specified compliance standard or regulation. You can also customize frameworks and controls to support internal audits with specific requirements.

Use the following links to get started with the Audit Manager API:

- **Actions**: An alphabetical list of all Audit Manager API operations.
- **Data types**: An alphabetical list of all Audit Manager data types.
- **Common parameters**: Parameters that all operations can use.
- **Common errors**: Client and server errors that all operations can return.

If you’re new to Audit Manager, we recommend that you review the Audit Manager User Guide.

**Usage**

```python
auditmanager(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

**Arguments**

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**: An alphabetical list of all Audit Manager API operations.
  - **Data types**: An alphabetical list of all Audit Manager data types.
  - **Common parameters**: Parameters that all operations can use.
  - **Common errors**: Client and server errors that all operations can return.

If you’re new to Audit Manager, we recommend that you review the Audit Manager User Guide.

### Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**: An alphabetical list of all Audit Manager API operations.
  - **Data types**: An alphabetical list of all Audit Manager data types.
  - **Common parameters**: Parameters that all operations can use.
  - **Common errors**: Client and server errors that all operations can return.

If you’re new to Audit Manager, we recommend that you review the Audit Manager User Guide.
- `secret_access_key`: AWS secret access key
- `session_token`: AWS temporary session token

- `profile`: The name of a profile to use. If not given, then the default profile is used.
- `anonymous`: Set anonymous credentials.

`endpoint` Optional shorthand for complete URL to use for the constructed client.

`region` Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- auditmanager(
  config = list(  
    credentials = list(  
      creds = list(  
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(  
    creds = list(  
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```
Operations
associate_assessment_report_evidence_folder
batch_associate_assessment_report_evidence
batch_create_delegation_by_assessment
batch_delete_delegation_by_assessment
batch_disassociate_assessment_report_evidence
batch_import_evidence_to_assessment_control
create_assessment
create_assessment_framework
create_assessment_report
delete_assessment
delete_assessment_framework
delete_assessment_framework_share
delete_assessment_report
delete_control
deregister_account
deregister_organization_admin_account
disassociate_assessment_report_evidence_folder
get_account_status
get_assessment
get_assessment_framework
get_assessment_report_url
get_change_logs
get_control
get_delegations
get_evidence
get_evidence_by_evidence_folder
get_evidence_file_upload_url
get_evidence_folder
get_evidence_folders_by_assessment
get_evidence_folders_by_assessment_control
get_insights
get_insights_by_assessment
get_organization_admin_account
get_services_in_scope
get_settings
list_assessment_control_insights_by_control_domain
list_assessment_control_insights_by_control_domain
list_assessment_control_insights_by_assessment
list_assessment_insights_by_control_domain
list_assessment_framework
list_assessment_framework_share_requests
list_assessment_reports
list_assessments
list_control_domain_insights
list_control_domain_insights_by_assessment
list_control_insights
list_keywords_for_data_source
list_notifications
list_tags_for_resource

Associates an evidence folder to an assessment report in an Audit Manager
Associates a list of evidence to an assessment report in an Audit Manager
Creates a batch of delegations for an assessment in Audit Manager
Deletes a batch of delegations for an assessment in Audit Manager
Disassociates a list of evidence from an assessment report in Audit Manager
Adds one or more pieces of evidence to a control in an Audit Manager
Creates an assessment in Audit Manager
Creates a custom framework in Audit Manager
Creates an assessment report for the specified assessment
Creates a new custom control in Audit Manager
Deletes an assessment in Audit Manager
Deletes a custom framework in Audit Manager
Deletes a share request for a custom framework in Audit Manager
Deletes an assessment report in Audit Manager
Deletes a custom control in Audit Manager
Deregisters an account in Audit Manager
Removes the specified Amazon Web Services account as a delegated administrator for Audit Manager
Disassociates an evidence folder from the specified assessment report
Gets the registration status of an account in Audit Manager
Gets information about a specified assessment
Gets information about a specified framework
Gets the URL of an assessment report in Audit Manager
Gets a list of changelogs from Audit Manager
Gets information about a specified control
Gets a list of delegations from an audit owner to a delegate
Gets information about a specified evidence item
Gets all evidence from a specified evidence folder in Audit Manager
Creates a presigned Amazon S3 URL that can be used to upload a file
Gets an evidence folder from a specified assessment in Audit Manager
Gets the evidence folders from a specified assessment in Audit Manager
Gets a list of evidence folders that are associated with a specified control
Gets the latest analytics data for all your current active assessments
Gets the latest analytics data for a specific active assessment
Gets the name of the delegated Amazon Web Services administrator for an account
Gets a list of all the Amazon Web Services that you can choose to include in your assessment
Gets the settings for a specified Amazon Web Services account
Lists the latest analytics data for controls within a specific control domain
Returns a list of the frameworks that are available in the Audit Manager
Returns a list of sent or received share requests for custom frameworks
Returns a list of assessment reports created in Audit Manager
Returns a list of current and past assessments from Audit Manager
Lists the latest analytics data for control domains across all of your active assessments
Lists analytics data for control domains within a specified active assessment
Returns a list of controls from Audit Manager
Returns a list of keywords that are pre-mapped to the specified control
Returns a list of all Audit Manager notifications
Returns a list of tags for the specified resource in Audit Manager
register_account
register_organization_admin_account
start_assessment_framework_share
tag_resource
untag_resource
update_assessment
update_assessment_control
update_assessment_control_set_status
update_assessment_framework
update_assessment_framework_share
update_assessment_status
update_control
update_settings
validate_assessment_report_integrity

Enables Audit Manager for the specified Amazon Web Services account
Enables an Amazon Web Services account within the organization as
Creates a share request for a custom framework in Audit Manager
Tags the specified resource in Audit Manager
Removes a tag from a resource in Audit Manager
Edits an Audit Manager assessment
Updates a control within an assessment in Audit Manager
Updates the status of a control set in an Audit Manager assessment
Updates a custom framework in Audit Manager
Updates a share request for a custom framework in Audit Manager
Updates the status of an assessment in Audit Manager
Updates a custom control in Audit Manager
Updates Audit Manager settings for the current account
Validates the integrity of an assessment report in Audit Manager

Examples

```r
## Not run:
svc <- auditmanager()
svc$associate_assessment_report_evidence_folder(
  Foo = 123
)

## End(Not run)
```

Amazon Augmented AI Runtime

Description

Amazon Augmented AI (Amazon A2I) adds the benefit of human judgment to any machine learning application. When an AI application can’t evaluate data with a high degree of confidence, human reviewers can take over. This human review is called a human review workflow. To create and start a human review workflow, you need three resources: a worker task template, a flow definition, and a human loop.

For information about these resources and prerequisites for using Amazon A2I, see Get Started with Amazon Augmented AI in the Amazon SageMaker Developer Guide.

This API reference includes information about API actions and data types that you can use to interact with Amazon A2I programmatically. Use this guide to:

- Start a human loop with the `start_human_loop` operation when using Amazon A2I with a custom task type. To learn more about the difference between custom and built-in task types, see Use Task Types. To learn how to start a human loop using this API, see Create and Start a Human Loop for a Custom Task Type in the Amazon SageMaker Developer Guide.
• Manage your human loops. You can list all human loops that you have created, describe individual human loops, and stop and delete human loops. To learn more, see Monitor and Manage Your Human Loop in the Amazon SageMaker Developer Guide.

Amazon A2I integrates APIs from various AWS services to create and start human review workflows for those services. To learn how Amazon A2I uses these APIs, see Use APIs in Amazon A2I in the Amazon SageMaker Developer Guide.

Usage

```r
augmentedairuntime(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- `config`: Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
    - `creds`:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
    - `anonymous`: Set anonymous credentials.
  - `endpoint`: The complete URL to use for the constructed client.
  - `region`: The AWS Region used in instantiating the client.
  - `close_connection`: Immediately close all HTTP connections.
  - `timeout`: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - `s3_force_path_style`: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

- `credentials`: Optional credentials shorthand for the `config` parameter
  - `creds`:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - `profile`: The name of a profile to use. If not given, then the default profile is used.
  - `anonymous`: Set anonymous credentials.
endpoint  Optional shorthand for complete URL to use for the constructed client.
region    Optional shorthand for AWS Region used in instantiating the client.

Value
A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```
svc <- augmentedairuntime(
  config = list(  
    credentials = list(  
      creds = list(  
        access_key_id = "string",  
        secret_access_key = "string",  
        session_token = "string"  
      ),  
      profile = "string",  
      anonymous = "logical"  
    ),  
    endpoint = "string",  
    region = "string",  
    close_connection = "logical",  
    timeout = "numeric",  
    s3_force_path_style = "logical",  
    sts_regional_endpoint = "string"  
  ),  
  credentials = list(  
    creds = list(  
      access_key_id = "string",  
      secret_access_key = "string",  
      session_token = "string"  
    ),  
    profile = "string",  
    anonymous = "logical"  
  ),  
  endpoint = "string",  
  region = "string"  
)
```

Operations

- `delete_human_loop` Deletes the specified human loop for a flow definition
- `describe_human_loop` Returns information about the specified human loop
- `list_human_loops` Returns information about human loops, given the specified parameters
- `start_human_loop` Starts a human loop, provided that at least one activation condition is met
- `stop_human_loop` Stops the specified human loop
Example

```r
## Not run:
svc <- augmentedairuntime()
svc$delete_human_loop(
  Foo = 123
)

## End(Not run)
```

Description

Amazon EC2 Auto Scaling

Amazon EC2 Auto Scaling is designed to automatically launch and terminate EC2 instances based on user-defined scaling policies, scheduled actions, and health checks.

For more information, see the Amazon EC2 Auto Scaling User Guide and the Amazon EC2 Auto Scaling API Reference.

Usage

```r
autoscaling(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
• **close_connection**: Immediately close all HTTP connections.

• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

• **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.

• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-entities.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-entities.html)

### credentials
Optional credentials shorthand for the config parameter

• **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

### endpoint
Optional shorthand for complete URL to use for the constructed client.

### region
Optional shorthand for AWS Region used in instantiating the client.

### Value
A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

### Service syntax

```r
svc <- autoscaling(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(  
```
creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

attach_instances
attach_launch_load_balancers
attach_launch_balancer_target_groups
attach_traffic_sources
batch_delete_scheduled_action
batch_put_scheduled_update_group_action
cancel_instance_refresh
complete_lifecycle_action
create_auto_scaling_group
create_launch_configuration
create_or_update_tags
delete_auto_scaling_group
delete_launch_configuration
delete_lifecycle_hook
delete_notification_configuration
delete_policy
delete_scheduled_action
delete_tags
delete_warm_pool
describe_account_limits
describe_adjustment_types
describe_auto_scaling_groups
describe_auto_scaling_instances
describe_auto_scaling_notification_types
describe_instance_refreshes
describe_launch_configurations
describe_life_cycle_hows
describe_life_cycle_hook_types
describe_load_balancers
describe_load_balancer_target_groups
describe_metric_collection_types
describe_notification_configurations
describe_policies
describe_scaling_activities

Attaches one or more EC2 instances to the specified Auto Scaling group
This API operation is superseded by AttachTrafficSources, which can attach multiple instances
Attaches one or more traffic sources to the specified Auto Scaling group
Deletes one or more scheduled actions for the specified Auto Scaling group
Creates or updates one or more scheduled scaling actions for an Auto Scaling group
Cancels an instance refresh or rollback that is in progress
Completes the lifecycle action for the specified token or instance with the specified result
Creates a launch configuration
Creates or updates tags for the specified Auto Scaling group
Deletes the specified Auto Scaling group
Deletes the specified launch configuration
Deletes the specified lifecycle hook
Deletes the specified notification
Deletes the specified scaling policy
Deletes the specified scheduled action
Deletes the specified tags
Deletes the warm pool for the specified Auto Scaling group
Describes the current Amazon EC2 Auto Scaling resource quotas for your account
Describes the available adjustment types for step scaling and simple scaling policies
Gets information about the Auto Scaling groups in the account and Region
Gets information about the Auto Scaling instances in the account and Region
Describes the notification types that are supported by Amazon EC2 Auto Scaling
Gets information about the instance refreshes for the specified Auto Scaling group
Gets information about the launch configurations in the account and Region
Gets information about the lifecycle hooks for the specified Auto Scaling group
Describes the available types of lifecycle hooks
This API operation is superseded by DescribeTrafficSources, which can describe multiple traffic sources
This API operation is superseded by DescribeTrafficSources, which can describe multiple traffic sources
Describes the available CloudWatch metrics for Amazon EC2 Auto Scaling
Gets information about the Amazon SNS notifications that are configured for one or more Auto Scaling groups
Gets information about the scaling policies in the account and Region
Gets information about the scaling activities in the account and Region
describe_scaling_process_types
describe_scheduled_actions
describe_tags
describe_termination_policy_types
describe_traffic_sources
describe_warm_pool
detach_instances
detach_load_balancers
detach_load_balancer_target_groups
detach_traffic_sources
disable_metrics_collection
enable_metrics_collection
enter_standby
execute_policy
exit_standby
get_predictive_scaling_forecast
put_lifecycle_hook
put_notification_configuration
put_scaling_policy
put_scheduled_update_group_action
put_warm_pool
record_lifecycle_action_heartbeat
resume_processes
rollback_instance_refresh
set_desired_capacity
set_instance_health
set_instance_protection
start_instance_refresh
suspend_processes
terminate_instance_in_auto_scaling_group
update_auto_scaling_group

describes the scaling process types for use with the ResumeProcesses and SuspendProcesses APIs.
Gets information about the scheduled actions that haven’t run or that have not reached their end time.
Describes the specified tags.
Describes the termination policies supported by Amazon EC2 Auto Scaling.
Gets information about the traffic sources for the specified Auto Scaling group.
Gets information about a warm pool and its instances.
Removes one or more instances from the specified Auto Scaling group.
This API operation is superseded by DetachTrafficSources, which can detach multiple traffic sources.
This API operation is superseded by DetachTrafficSources, which can detach multiple traffic sources.
Detaches one or more traffic sources from the specified Auto Scaling group.
Disables group metrics collection for the specified Auto Scaling group.
Enables group metrics collection for the specified Auto Scaling group.
Moves the specified instances into the standby state.
Executes the specified policy.
Moves the specified instances out of the standby state.
Retrieves the forecast data for a predictive scaling policy.
Creates or updates a lifecycle hook for the specified Auto Scaling group.
Configures an Auto Scaling group to send notifications when specified events take place.
Creates or updates a scaling policy for an Auto Scaling group.
Creates or updates a scheduled scaling action for an Auto Scaling group.
Creates or updates a warm pool for the specified Auto Scaling group.
Records a heartbeat for the lifecycle action associated with the specified token.
Resumes the specified suspended auto scaling processes, or all suspended processes.
Cancels an instance refresh that is in progress and rolls back any changes that it made.
Sets the size of the specified Auto Scaling group.
Sets the health status of the specified instance.
Updates the instance protection settings of the specified instances.
Starts an instance refresh.
Suspends the specified auto scaling processes, or all processes, for the specified Auto Scaling group.
Terminates the specified instance and optionally adjusts the desired group size.

We strongly recommend that all Auto Scaling groups use launch templates to ensure full functionality.
**Description**

AWS Auto Scaling

Use AWS Auto Scaling to create scaling plans for your applications to automatically scale your scalable AWS resources.

**API Summary**

You can use the AWS Auto Scaling service API to accomplish the following tasks:

- Create and manage scaling plans
- Define target tracking scaling policies to dynamically scale your resources based on utilization
- Scale Amazon EC2 Auto Scaling groups using predictive scaling and dynamic scaling to scale your Amazon EC2 capacity faster
- Set minimum and maximum capacity limits
- Retrieve information on existing scaling plans
- Access current forecast data and historical forecast data for up to 56 days previous

To learn more about AWS Auto Scaling, including information about granting IAM users required permissions for AWS Auto Scaling actions, see the AWS Auto Scaling User Guide.

**Usage**

```r
autoscalingplans(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

**Arguments**

- `config`: Optional configuration of credentials, endpoint, and/or region.
  
  - `credentials`:
    - `creds`:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
      - `anonymous`: Set anonymous credentials.
    - `endpoint`: The complete URL to use for the constructed client.
• **region**: The AWS Region used in instantiating the client.
• **close_connection**: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

**credentials**

Optional credentials shorthand for the config parameter

• **creds**: 
  – **access_key_id**: AWS access key ID
  – **secret_access_key**: AWS secret access key
  – **session_token**: AWS temporary session token
• **profile**: The name of a profile to use. If not given, then the default profile is used.
• **anonymous**: Set anonymous credentials.

**endpoint**

Optional shorthand for complete URL to use for the constructed client.

**region**

Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- autoscalingplans(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
)```
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

create_scaling_plan Creates a scaling plan
delete_scaling_plan Deletes the specified scaling plan
describe_scaling_plan_resources Describes the scalable resources in the specified scaling plan
describe_scaling_plans Describes one or more of your scaling plans
get_scaling_plan_resource_forecast_data Retrieves the forecast data for a scalable resource
update_scaling_plan Updates the specified scaling plan

Examples

## Not run:
svc <- autoscalingplans()
svc$create_scaling_plan(
    Foo = 123
)

## End(Not run)

backup

AWS Backup

Description

Backup

Backup is a unified backup service designed to protect Amazon Web Services services and their associated data. Backup simplifies the creation, migration, restoration, and deletion of backups, while also providing reporting and auditing.

Usage

backup(config = list(), credentials = list(), endpoint = NULL, region = NULL)
Arguments

**config** Optional configuration of credentials, endpoint, and/or region.

- **credentials**:
  - **creds**:
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
    - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.
- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.

**credentials** Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- backup(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
```
backup

    secret_access_key = "string",
    session_token = "string"

),
    profile = "string",
    anonymous = "logical"

),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"

),
    credentials = list(
        creds = list(
            access_key_id = "string",
            secret_access_key = "string",
            session_token = "string"
        ),
        profile = "string",
        anonymous = "logical"
    ),
    endpoint = "string",
    region = "string"
)

Operations

cancel_legal_hold This action removes the specified legal hold on a recovery point
create_backup_plan Creates a backup plan using a backup plan name and backup rules
create_backup_selection Creates a JSON document that specifies a set of resources to assign to a backup plan
create_backup_vault Creates a logical container where backups are stored
create_framework Creates a framework with one or more controls
create_legally_air_gapped_backup_vault This request creates a logical container to where backups may be copied
create_report_plan Creates a report plan
create_restore_testing_plan This is the first of two steps to create a restore testing plan; once this request is successful, finish the procedure with request CreateRestoreTestingSelection
create_restore_testing_selection This request can be sent after CreateRestoreTestingPlan request returns successfully
delete_backup_plan Deletes a backup plan
delete_backup_selection Deletes the resource selection associated with a backup plan that is specified by the SelectionId
delete_backup_vault Deletes the backup vault identified by its name
delete_backup_vault_access_policy Deletes the policy document that manages permissions on a backup vault
delete_backup_vault_lock_configuration Deletes Backup Vault Lock from a backup vault specified by a backup vault name
delete_backup_vault_notifications Deletes event notifications for the specified backup vault
delete_framework Deletes the framework specified by a framework name
delete_recovery_point Deletes the recovery point specified by a recovery point ID
delete_report_plan Deletes the report plan specified by a report plan name
delete_restore_testing_plan This request deletes the specified restore testing plan
null
Examples

```r
## Not run:
svc <- backup()
svc$cancel_legal_hold(
  Foo = 123
)

## End(Not run)
```

backupgateway  AWS Backup Gateway

Description

Backup gateway

Backup gateway connects Backup to your hypervisor, so you can create, store, and restore backups of your virtual machines (VMs) anywhere, whether on-premises or in the VMware Cloud (VMC) on Amazon Web Services.
Add on-premises resources by connecting to a hypervisor through a gateway. Backup will automatically discover the resources in your hypervisor.

Use Backup to assign virtual or on-premises resources to a backup plan, or run on-demand backups. Once you have backed up your resources, you can view them and restore them like any resource supported by Backup.

To download the Amazon Web Services software to get started, navigate to the Backup console, choose Gateways, then choose Create gateway.

Usage

```python
backupgateway(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **creds**:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.
- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
• **anonymous**: Set anonymous credentials.

**endpoint**  
Optional shorthand for complete URL to use for the constructed client.

**region**  
Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- backupgateway(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

**Operations**

- `associate_gateway_to_server`  
  Associates a backup gateway with your server

- `create_gateway`  
  Creates a backup gateway

- `delete_gateway`  
  Deletes a backup gateway
Example

```r
## Not run:
svc <- backupgateway()
svc$associate_gateway_to_server(
  Foo = 123
)
## End(Not run)
```

### backupstorage

**Description**

The frontend service for Cryo Storage.

**Usage**

```r
backupstorage(
  config = list(),
)```
backupstorage

```python
credentials = list(),
endpoint = NULL,
region = NULL
```

**Arguments**

**config**

Optional configuration of credentials, endpoint, and/or region.

- **credentials**:
  - **creds**:
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
    - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.
- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

**credentials**

Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint**

Optional shorthand for complete URL to use for the constructed client.

**region**

Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

svc <- backupstorage(
    config = list(
        credentials = list(
            creds = list(
                access_key_id = "string",
                secret_access_key = "string",
                session_token = "string"
            ),
            profile = "string",
            anonymous = "logical"
        ),
        endpoint = "string",
        region = "string",
        close_connection = "logical",
        timeout = "numeric",
        s3_force_path_style = "logical",
        sts_regional_endpoint = "string"
    ),
    credentials = list(
        creds = list(
            access_key_id = "string",
            secret_access_key = "string",
            session_token = "string"
        ),
        profile = "string",
        anonymous = "logical"
    ),
    endpoint = "string",
    region = "string"
)

Operations

- delete_object: Delete Object from the incremental base Backup
- get_chunk: Gets the specified object’s chunk
- get_object_metadata: Get metadata associated with an Object
- list_chunks: List chunks in a given Object
- list_objects: List all Objects in a given Backup
- notify_object_complete: Complete upload
- put_chunk: Upload chunk
- put_object: Upload object that can store object metadata String and data blob in single API call using inline chunk
- start_object: Start upload containing one or many chunks
### Examples

```r
## Not run:
svc <- backupstorage()
svc$delete_object(
  Foo = 123
)
## End(Not run)
```

---

**batch**

**AWS Batch**

### Description

**Batch**

Using Batch, you can run batch computing workloads on the Amazon Web Services Cloud. Batch computing is a common means for developers, scientists, and engineers to access large amounts of compute resources. Batch uses the advantages of the batch computing to remove the undifferentiated heavy lifting of configuring and managing required infrastructure. At the same time, it also adopts a familiar batch computing software approach. You can use Batch to efficiently provision resources, and work toward eliminating capacity constraints, reducing your overall compute costs, and delivering results more quickly.

As a fully managed service, Batch can run batch computing workloads of any scale. Batch automatically provisions compute resources and optimizes workload distribution based on the quantity and scale of your specific workloads. With Batch, there’s no need to install or manage batch computing software. This means that you can focus on analyzing results and solving your specific problems instead.

### Usage

```r
batch(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

### Arguments

- **config**
  - Optional configuration of credentials, endpoint, and/or region.
  - **credentials:**
    - **creds:**
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - **profile:** The name of a profile to use. If not given, then the default profile is used.
    - **anonymous:** Set anonymous credentials.
  - **endpoint:** The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

**credentials**

Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint**

Optional shorthand for complete URL to use for the constructed client.

**region**

Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- batch(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
)```

credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

cancel_job                                                     Cancels a job in an Batch job queue
create_compute_environment                                     Creates an Batch compute environment
create_job_queue                                              Creates an Batch job queue
create_scheduling_policy                                      Creates an Batch scheduling policy
delete_compute_environment                                    Deletes an Batch compute environment
delete_job_queue                                              Deletes the specified job queue
delete_scheduling_policy                                      Deletes the specified scheduling policy
deregister_job_definition                                     Deregisters an Batch job definition
describe_compute_environments                                 Describes one or more of your compute environments
describe_job_definitions                                      Describes a list of job definitions
describe_job_queues                                           Describes one or more of your job queues
describe_jobs                                                 Describes a list of Batch jobs
describe_scheduling_policies                                 Describes one or more of your scheduling policies
list_jobs                                                      Returns a list of Batch jobs
list_scheduling_policies                                     Returns a list of Batch scheduling policies
list_tags_for_resource                                      Lists the tags for an Batch resource
register_job_definition                                       Registers an Batch job definition
submit_job                                                     Submits an Batch job from a job definition
tag_resource                                                  Associates the specified tags to a resource with the specified resourceArn
terminate_job                                                 Terminates a job in a job queue
untag_resource                                                 Deletes specified tags from an Batch resource
update_compute_environment                                    Updates an Batch compute environment
update_job_queue                                              Updates a job queue
update_scheduling_policy                                      Updates a scheduling policy

Examples

## Not run:
svc <- batch()
# This example cancels a job with the specified job ID.
svc$cancel_job(
jobId = "1d828f65-7a4d-42e8-996d-3b900ed59dc4",
reason = "Cancelling job."
}

## End(Not run)

---

bedrock

Amazon Bedrock

**Description**

Describes the API operations for creating, managing, fine-turning, and evaluating Amazon Bedrock models.

**Usage**

```
bedrock(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

**Arguments**

- **config**
  - Optional configuration of credentials, endpoint, and/or region.
  - **credentials**
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

- **credentials**
  - Optional credentials shorthand for the config parameter
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
- **session_token**: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.

- **anonymous**: Set anonymous credentials.

**endpoint**  
Optional shorthand for complete URL to use for the constructed client.

**region**  
Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- bedrock(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

**Operations**
create_evaluation_job
create_guardrail
create_guardrail_version
create_model_customization_job
create_provisioned_model_throughput
delete_custom_model
delete_guardrail
delete_model_invocation_logging_configuration
delete_provisioned_model_throughput
get_custom_model
get_evaluation_job
get_foundation_model
get_guardrail
get_model_customization_job
get_model_invocation_logging_configuration
get_provisioned_model_throughput
list_custom_models
list_evaluation_jobs
list_foundation_models
list_guards
list_model_customization_jobs
list_provisioned_model_throughputs
list_tags_for_resource
put_model_invocation_logging_configuration
stop_evaluation_job
stop_model_customization_job
tag_resource
untag_resource
update_guardrail
update_provisioned_model_throughput

API operation for creating and managing Amazon Bedrock automatic model evaluation jobs.

create_guardrail
Creates a guardrail to block topics and to filter out harmful content.

create_guardrail_version
Creates a version of the guardrail.

create_model_customization_job
Creates a fine-tuning job to customize a base model.

create_provisioned_model_throughput
Creates dedicated throughput for a base or custom model with the model units and for the duration that you specify.

delete_custom_model
Deletes a custom model that you created earlier.

delete_guardrail
Deletes a guardrail.

delete_model_invocation_logging_configuration
Deletes the invocation logging.

delete_provisioned_model_throughput
Deletes a Provisioned Throughput.

get_custom_model
Get the properties associated with a Amazon Bedrock custom model that you have created.

get_evaluation_job
Retrieves the properties associated with a model evaluation job, including the status of the job.

get_foundation_model
Gets details about a Amazon Bedrock foundation model.

get_guardrail
Retrieves the properties associated with a guardrail.

get_model_customization_job
Retrieves the properties associated with a model-customization job, including the status of the job.

get_model_invocation_logging_configuration
Gets the current configuration values for model invocation logging.

get_provisioned_model_throughput
Returns details for a Provisioned Throughput.

list_custom_models
Returns a list of the custom models that you have created with the CreateModelCustomizationJob operation.

list_evaluation_jobs
Lists model evaluation jobs.

list_foundation_models
Lists Amazon Bedrock foundation models that you can use.

list_guards
Lists details about all the guardrails in an account.

list_model_customization_jobs
Returns a list of model customization jobs that you have submitted.

list_provisioned_model_throughputs
Lists the Provisioned Throughputs in the account.

list_tags_for_resource
List the tags associated with the specified resource.

put_model_invocation_logging_configuration
Set the configuration values for model invocation logging.

stop_evaluation_job
 Stops an in progress model evaluation job.

stop_model_customization_job
 Stops an active model customization job.

tag_resource
 Associate tags with a resource.

untag_resource
 Remove one or more tags from a resource.

update_guardrail
 Updates a guardrail with the values you specify.

update_provisioned_model_throughput
 Updates the name or associated model for a Provisioned Throughput.

Examples

```r
## Not run:
svc <- bedrock()
svc$create_evaluation_job(
  Foo = 123
)

## End(Not run)
```
Description

Describes the API operations for running inference using Amazon Bedrock models.

Usage

```r
bedrockruntime(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

config

Optional configuration of credentials, endpoint, and/or region.

- **credentials**
  - creds:  
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
    - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.
- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

credentials

Optional credentials shorthand for the config parameter

- creds:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

endpoint

Optional shorthand for complete URL to use for the constructed client.

region

Optional shorthand for AWS Region used in instantiating the client.
Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- bedrockruntime(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `invoke_model` Invokes the specified Amazon Bedrock model to run inference using the prompt and inference parameters provided in the request body.
- `invoke_model_with_response_stream` Invoke the specified Amazon Bedrock model to run inference using the prompt and inference parameters provided in the request body.

Examples

```r
## Not run:
```
### Description

Amazon Web Services Billing Conductor is a fully managed service that you can use to customize a proforma version of your billing data each month, to accurately show or chargeback your end customers. Amazon Web Services Billing Conductor doesn’t change the way you’re billed by Amazon Web Services each month by design. Instead, it provides you with a mechanism to configure, generate, and display rates to certain customers over a given billing period. You can also analyze the difference between the rates you apply to your accounting groupings relative to your actual rates from Amazon Web Services. As a result of your Amazon Web Services Billing Conductor configuration, the payer account can also see the custom rate applied on the billing details page of the Amazon Web Services Billing console, or configure a cost and usage report per billing group.

This documentation shows how you can configure Amazon Web Services Billing Conductor using its API. For more information about using the Amazon Web Services Billing Conductor user interface, see the Amazon Web Services Billing Conductor User Guide.

### Usage

```r
billingconductor(config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL)
```

### Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**: 
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
billingconductor

- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials** Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- billingconductor(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  )
)
```
billingconductor

credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

associate_accounts
associate_pricing_rules
batch_associate_resources_to_custom_line_item
batch_disassociate_resources_from_custom_line_item
create_billing_group
create_custom_line_item
create_pricing_plan
create_pricing_rule
delete_billing_group
delete_custom_line_item
delete_pricing_plan
delete_pricing_rule
disassociate_accounts
disassociate_pricing_rules
get_billing_group_cost_report
list_account_associations
list_billing_group_cost_reports
list_billing_groups
list_custom_line_items
list_custom_line_item_versions
list_pricing_plans
list_pricing_plans_associated_with_pricing_rule
list_pricing_rules
list_pricing_rules_associated_to_pricing_plan
list_resources_associated_to_custom_line_item
list_tags_for_resource
tag_resource
untag_resource
update_billing_group
update_custom_line_item
update_pricing_plan
update_pricing_rule

Connects an array of account IDs in a consolidated billing family to a predefined billing group
Connects an array of PricingRuleArns to a defined PricingPlan
Associates a batch of resources to a percentage custom line item
Disassociates a batch of resources from a percentage custom line item
Creates a billing group that resembles a consolidated billing family that Amazon Web Services charges, based off of the predefined pricing plan computation
Creates a custom line item that can be used to create a one-time fixed charge that can be applied to a billing group
Creates a pricing plan that is used for computing Amazon Web Services charges for billing groups
Connects an array of PricingRuleArns to a defined PricingPlan
Deletes a billing group
Deletes the custom line item identified by the given ARN in the current or previous billing period
Deletes a pricing plan
Deletes the pricing rule that’s identified by the input Amazon Resource Name (ARN)
Removes the specified list of account IDs from the given billing group
Disassociates a list of pricing rules from a pricing plan
Retrieves the margin summary report, which includes the Amazon Web Services actual charges and the calculated Amazon Web Services charges based on the associated pricing plan of a billing group
This is a paginated call to list linked accounts that are linked to the payer account for the specified time period
A paginated call to retrieve a summary report of actual Amazon Web Services charges for a billing group
A paginated call to get a list of billing groups for the given billing period
A paginated call to get a list of all custom line items (FFLIs) for the given billing group
A paginated call to get a list of all custom line item versions
A paginated call to get pricing plans for the given billing period
A list of the pricing plans that are associated with a pricing rule
Describes a pricing rule that can be associated to a pricing plan, or set of plans
Lists the pricing rules that are associated with a pricing plan
List the resources that are associated to a custom line item
A list the tags for a resource
Associates the specified tags to a resource with the specified resourceArn
Deletes specified tags from a resource
This updates an existing billing group
Update an existing custom line item in the current or previous billing period
This updates an existing pricing plan
Updates an existing pricing rule
Examples

```r
## Not run:
svc <- billingconductor()
svc$associate_accounts(
  Foo = 123
)
## End(Not run)
```

---

braket                Braket

Description

The Amazon Braket API Reference provides information about the operations and structures supported in Amazon Braket.

Additional Resources:

- Amazon Braket Developer Guide

Usage

```r
braket(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy
  

  **credentials**: Optional credentials shorthand for the config parameter

  • **creds**:
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
    - **session_token**: AWS temporary session token
  
  • **profile**: The name of a profile to use. If not given, then the default profile is used.

  • **anonymous**: Set anonymous credentials.

  **endpoint**: Optional shorthand for complete URL to use for the constructed client.

  **region**: Optional shorthand for AWS Region used in instantiating the client.

  **Value**

  A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

  **Service syntax**

  ```
  svc <- braket(
    config = list(
      credentials = list(
        creds = list(
          access_key_id = "string",
          secret_access_key = "string",
          session_token = "string"
        ),
        profile = "string",
        anonymous = "logical"
      ),
      endpoint = "string",
      region = "string",
      close_connection = "logical",
      timeout = "numeric",
      s3_force_path_style = "logical",
      sts_regional_endpoint = "string"
    ),
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
    
    
```
Use the Amazon Web Services Budgets API to plan your service usage, service costs, and instance reservations. This API reference provides descriptions, syntax, and usage examples for each of the actions and data types for the Amazon Web Services Budgets feature.

Budgets provide you with a way to see the following information:

- How close your plan is to your budgeted amount or to the free tier limits
- Your usage-to-date, including how much you’ve used of your Reserved Instances (RIs)
• Your current estimated charges from Amazon Web Services, and how much your predicted usage will accrue in charges by the end of the month
• How much of your budget has been used

Amazon Web Services updates your budget status several times a day. Budgets track your unblended costs, subscriptions, refunds, and RIs. You can create the following types of budgets:

• **Cost budgets** - Plan how much you want to spend on a service.
• **Usage budgets** - Plan how much you want to use one or more services.
• **RI utilization budgets** - Define a utilization threshold, and receive alerts when your RI usage falls below that threshold. This lets you see if your RIs are unused or under-utilized.
• **RI coverage budgets** - Define a coverage threshold, and receive alerts when the number of your instance hours that are covered by RIs fall below that threshold. This lets you see how much of your instance usage is covered by a reservation.

Service Endpoint

The Amazon Web Services Budgets API provides the following endpoint:

• https://budgets.amazonaws.com

For information about costs that are associated with the Amazon Web Services Budgets API, see Amazon Web Services Cost Management Pricing.

Usage

    budgets(config = list(), credentials = list(), endpoint = NULL, region = NULL)

Arguments

    config Optional configuration of credentials, endpoint, and/or region.
    
    • credentials:
        – creds:
            ◆ access_key_id: AWS access key ID
            ◆ secret_access_key: AWS secret access key
            ◆ session_token: AWS temporary session token
        – profile: The name of a profile to use. If not given, then the default profile is used.
        – anonymous: Set anonymous credentials.
    • endpoint: The complete URL to use for the constructed client.
    • region: The AWS Region used in instantiating the client.
    • close_connection: Immediately close all HTTP connections.
    • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
    • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
    • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html
credentials

Optional credentials shorthand for the config parameter

• creds:
  – access_key_id: AWS access key ID
  – secret_access_key: AWS secret access key
  – session_token: AWS temporary session token

• profile: The name of a profile to use. If not given, then the default profile is used.

• anonymous: Set anonymous credentials.

document

Optional credentials shorthand for the config parameter

• creds:
  – access_key_id: AWS access key ID
  – secret_access_key: AWS secret access key
  – session_token: AWS temporary session token

• profile: The name of a profile to use. If not given, then the default profile is used.

• anonymous: Set anonymous credentials.

document

Optional credentials shorthand for the config parameter

• creds:
  – access_key_id: AWS access key ID
  – secret_access_key: AWS secret access key
  – session_token: AWS temporary session token

• profile: The name of a profile to use. If not given, then the default profile is used.

• anonymous: Set anonymous credentials.

document

Optional credentials shorthand for the config parameter

• creds:
  – access_key_id: AWS access key ID
  – secret_access_key: AWS secret access key
  – session_token: AWS temporary session token

• profile: The name of a profile to use. If not given, then the default profile is used.

• anonymous: Set anonymous credentials.

document

Optional credentials shorthand for the config parameter

• creds:
  – access_key_id: AWS access key ID
  – secret_access_key: AWS secret access key
  – session_token: AWS temporary session token

• profile: The name of a profile to use. If not given, then the default profile is used.

• anonymous: Set anonymous credentials.

document

Optional credentials shorthand for the config parameter

• creds:
  – access_key_id: AWS access key ID
  – secret_access_key: AWS secret access key
  – session_token: AWS temporary session token

• profile: The name of a profile to use. If not given, then the default profile is used.

• anonymous: Set anonymous credentials.

document

Optional credentials shorthand for the config parameter

• creds:
  – access_key_id: AWS access key ID
  – secret_access_key: AWS secret access key
  – session_token: AWS temporary session token

• profile: The name of a profile to use. If not given, then the default profile is used.

• anonymous: Set anonymous credentials.

document

Optional credentials shorthand for the config parameter

• creds:
  – access_key_id: AWS access key ID
  – secret_access_key: AWS secret access key
  – session_token: AWS temporary session token

• profile: The name of a profile to use. If not given, then the default profile is used.

• anonymous: Set anonymous credentials.
region = "string"
)

Operations

create_budget
create_budget_action
create_notification
create_subscriber
delete_budget
delete_budget_action
delete_notification
delete_subscriber
describe_budget
describe_budget_action
describe_budget_action_histories
describe_budget_actions_for_account
describe_budget_actions_for_budget
describe_budget_notifications_for_account
describe_budget_performance_history
describe_budgets
describe_notifications_for_budget
describe_subscribers_for_notification
execute_budget_action
update_budget
update_budget_action
update_notification
update_subscriber

Creates a budget and, if included, notifications and subscribers
Creates a budget action
Creates a notification
Creates a subscriber
Deletes a budget
Deletes a budget action
Deletes a notification
Deletes a subscriber
Describes a budget
Describes a budget action detail
Describes a budget action history detail
Describes all of the budget actions for an account
Describes all of the budget actions for a budget
Lists the budget names and notifications that are associated with an account
Describes the history for DAILY, MONTHLY, and QUARTERLY budgets
Lists the budgets that are associated with an account
Lists the notifications that are associated with a budget
Lists the subscribers that are associated with a notification
Executes a budget action
Updates a budget
Updates a budget action
Updates a notification
Updates a subscriber

Examples

## Not run:
svc <- budgets()
svc$create_budget(
  Foo = 123
)

## End(Not run)
Description

Cloud9

Cloud9 is a collection of tools that you can use to code, build, run, test, debug, and release software in the cloud.

For more information about Cloud9, see the Cloud9 User Guide.

Cloud9 supports these operations:

- **create_environment_ec2**: Creates an Cloud9 development environment, launches an Amazon EC2 instance, and then connects from the instance to the environment.
- **create_environment_membership**: Adds an environment member to an environment.
- **delete_environment**: Deletes an environment. If an Amazon EC2 instance is connected to the environment, also terminates the instance.
- **delete_environment_membership**: Deletes an environment member from an environment.
- **describe_environment_memberships**: Gets information about environment members for an environment.
- **describe_environments**: Gets information about environments.
- **describe_environment_status**: Gets status information for an environment.
- **list_environments**: Gets a list of environment identifiers.
- **list_tags_for_resource**: Gets the tags for an environment.
- **tag_resource**: Adds tags to an environment.
- **untag_resource**: Removes tags from an environment.
- **update_environment**: Changes the settings of an existing environment.
- **update_environment_membership**: Changes the settings of an existing environment member for an environment.

Usage

```r
cloud9(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
cloud9

- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy `https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html`

**credentials**

Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint**

Optional shorthand for complete URL to use for the constructed client.

**region**

Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- cloud9(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
```
creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

### Operations

- **create_environment_ec2**: Creates an Cloud9 development environment, launches an Amazon Elastic Compute Cloud (Amazon EC2) instance, and then connects from the instance to the environment.
- **create_environment_membership**: Adds an environment member to an Cloud9 development environment.
- **delete_environment**: Deletes an Cloud9 development environment.
- **delete_environment_membership**: Deletes an environment member from a development environment.
- **describe_environment_memberships**: Gets information about environment members for an Cloud9 development environment.
- **describe_environments**: Gets a list of Cloud9 development environment identifiers.
- **describe_environment_status**: Gets status information for an Cloud9 development environment.
- **list_environments**: Gets a list of the tags associated with an Cloud9 development environment.
- **list_tags_for_resource**: Adds tags to an Cloud9 development environment.
- **tag_resource**: Removes tags from an Cloud9 development environment.
- **untag_resource**: Changes the settings of an existing Cloud9 development environment.
- **update_environment**: Changes the settings of an existing environment member for an Cloud9 development environment.

### Examples

```r
## Not run:
svc <- cloud9()
#
svc$create_environment_ec2(
    name = "my-demo-environment",
    automaticStopTimeMinutes = 60L,
    description = "This is my demonstration environment.",
    imageId = "amazonlinux-2023-x86_64",
    instanceType = "t2.micro",
    ownerArn = "arn:aws:iam::123456789012:user/MyDemoUser",
    subnetId = "subnet-6300cd1b"
)
## End(Not run)
```
cloudcontrolapi  AWS Cloud Control API

Description

For more information about Amazon Web Services Cloud Control API, see the Amazon Web Services Cloud Control API User Guide.

Usage

cloudcontrolapi(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

cfg    Optional configuration of credentials, endpoint, and/or region.

- credentials:
  - creds:
    - access_key_id: AWS access key ID
    - secret_access_key: AWS secret access key
    - session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.
- endpoint: The complete URL to use for the constructed client.
- region: The AWS Region used in instantiating the client.
- close_connection: Immediately close all HTTP connections.
- timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
- sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials    Optional credentials shorthand for the config parameter

- creds:
  - access_key_id: AWS access key ID
  - secret_access_key: AWS secret access key
  - session_token: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**
A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
cvc <- cloudcontrolapi(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

**Operations**

- `cancel_resource_request` Cancels the specified resource operation request
**clouddirectory**

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>create_resource</code></td>
<td>Creates the specified resource</td>
</tr>
<tr>
<td><code>delete_resource</code></td>
<td>Deletes the specified resource</td>
</tr>
<tr>
<td><code>get_resource</code></td>
<td>Returns the current state of the specified resource</td>
</tr>
<tr>
<td><code>get_resource_request_status</code></td>
<td>Returns the current status of a resource operation request</td>
</tr>
<tr>
<td><code>list_resource_requests</code></td>
<td>Returns existing resource operation requests</td>
</tr>
<tr>
<td><code>list_resources</code></td>
<td>Returns information about the specified resources</td>
</tr>
<tr>
<td><code>update_resource</code></td>
<td>Updates the specified property values in the resource</td>
</tr>
</tbody>
</table>

**Examples**

```r
## Not run:
svc <- cloudcontrolapi()
svc$cancel_resource_request(
  Foo = 123
)
## End(Not run)
```

---

**clouddirectory**  
*Amazon CloudDirectory*

**Description**

Amazon Cloud Directory

Amazon Cloud Directory is a component of the AWS Directory Service that simplifies the development and management of cloud-scale web, mobile, and IoT applications. This guide describes the Cloud Directory operations that you can call programmatically and includes detailed information on data types and errors. For information about Cloud Directory features, see [AWS Directory Service](https://aws.amazon.com/directory-service/) and the *Amazon Cloud Directory Developer Guide*.

**Usage**

```r
cloddirectory(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

**Arguments**

- `config`  
  Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:  
    - `creds`:  

* `access_key_id`: AWS access key ID
* `secret_access_key`: AWS secret access key
* `session_token`: AWS temporary session token

- `profile`: The name of a profile to use. If not given, then the default profile is used.
- `anonymous`: Set anonymous credentials.

- `endpoint`: The complete URL to use for the constructed client.
- `region`: The AWS Region used in instantiating the client.
- `close_connection`: Immediately close all HTTP connections.
- `timeout`: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- `s3_force_path_style`: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
- `sts_regional_endpoint`: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ensemble.html

**credentials**

Optional credentials shorthand for the `config` parameter

- `creds`:
  - `access_key_id`: AWS access key ID
  - `secret_access_key`: AWS secret access key
  - `session_token`: AWS temporary session token
  - `profile`: The name of a profile to use. If not given, then the default profile is used.
  - `anonymous`: Set anonymous credentials.

**endpoint**

Optional shorthand for complete URL to use for the constructed client.

**region**

Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
doctest
svc <- clouddirectory(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
```
clouddirectory

),
  endpoint = "string",
  region = "string",
  close_connection = "logical",
  timeout = "numeric",
  s3_force_path_style = "logical",
  sts_regional_endpoint = "string"
),
credentials = list(
  creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
  ),
  profile = "string",
  anonymous = "logical"
),
  endpoint = "string",
  region = "string"
)

Operations

  add_facet_to_object Adds a new Facet to an object
  apply_schema Copies the input published schema, at the specified version, into the Directory with the same name and version as that of the published schema
  attach_object Attaches an existing object to another object
  attach_policy Attaches a policy object to a regular object
  attach_to_index Attaches the specified object to the specified index
  attach_typed_link Attaches a typed link to a specified source and target object
  batch_read Performs all the read operations in a batch
  batch_write Performs all the write operations in a batch
  create_directory Creates a Directory by copying the published schema into the directory
  create_facet Creates a new Facet in a schema
  create_index Creates an index object
  create_object Creates an object in a Directory
  create_schema Creates a new schema in a development state
  create_typed_link_facet Creates a TypedLinkFacet
  delete_directory Deletes a directory
  delete_facet Deletes a given Facet
  delete_object Deletes an object and its associated attributes
  delete_schema Deletes a given schema
  delete_typed_link_facet Deletes a TypedLinkFacet
  detach_from_index Detaches the specified object from the specified index
  detach_object Detaches a given object from the parent object
  detach_policy Detaches a policy from an object
  detach_typed_link Detaches a typed link from a specified source and target object
  disable_directory Disables the specified directory
  enable_directory Enables the specified directory
get_applied_schema_version  
get_directory  
get_facet  
get_link_attributes  
get_object_attributes  
get_object_information  
get_schema_as_json  
get_typed_link_facet_information  
list_applied_schema_arns  
list_attached_indices  
list_development_schema_arns  
list_directories  
list_facet_attributes  
list_facet_names  
list_incoming_typed_links  
list_index  
list.managed_schema_arns  
list_object_attributes  
list_object_children  
list_object_parent_paths  
list_object_parents  
list_object_policies  
list_outgoing_typed_links  
list_policy_attachments  
list_published_schema_arns  
list_tags_for_resource  
list_typed_link_facet_attributes  
list_typed_link_facet_names  
lookup_policy  
publish_schema  
put_schema_from_json  
remove_facet_from_object  
tag_resource  
untag_resource  
update_facet  
update_link_attributes  
update_object_attributes  
update_schema  
update_typed_link_facet  
upgrade_applied_schema  
upgrade_published_schema  

Returns current applied schema version ARN, including the minor version in use
Retrieves metadata about a directory
Gets details of the Facet, such as facet name, attributes, Rules, or ObjectType
Retrieves attributes that are associated with a typed link
Retrieves attributes within a facet that are associated with an object
Retrieves metadata about an object
Retrieves a JSON representation of the schema
Returns the identity attribute order for a specific TypedLinkFacet
Lists schema major versions applied to a directory
Lists indices attached to the specified object
Retrieves each Amazon Resource Name (ARN) of schemas in the development state
Lists directories created within an account
Retrieves attributes attached to the facet
Retrieves the names of facets that exist in a schema
Returns a paginated list of all the incoming TypedLinkSpecifier information for an object
Lists objects attached to the specified index
Lists the major version families of each managed schema
Lists all attributes that are associated with an object
Returns a paginated list of child objects that are associated with a given object
Retrieves all available parent paths for any object type such as node, leaf node, policy node
Lists parent objects that are associated with a given object in pagination fashion
Retrieves policies attached to an object in pagination fashion
Returns a paginated list of all the outgoing TypedLinkSpecifier information for an object
Returns all of the ObjectIdentifiers to which a given policy is attached
Lists the major version families of each published schema
Returns tags for a resource
Returns a paginated list of all attribute definitions for a particular TypedLinkFacet
Returns a paginated list of TypedLink facet names for a particular schema
Lists all policies from the root of the Directory to the object specified
Publishes a development schema with a major version and a recommended minor version
Allows a schema to be updated using JSON upload
Removes the specified facet from the specified object
An API operation for adding tags to a resource
An API operation for removing tags from a resource
Does the following:
Updates a given typed link’s attributes
Updates a given object’s attributes
Updates the schema name with a new name
Updates a TypedLinkFacet
Upgrades a single directory in-place using the PublishedSchemaArn with schema updates
Upgrades a published schema under a new minor version revision using the current context

Examples

```r
## Not run:
svc <- clouddirectory()
svc$add_facet_to_object(
```
### Description

CloudFormation allows you to create and manage Amazon Web Services infrastructure deployments predictably and repeatedly. You can use CloudFormation to leverage Amazon Web Services products, such as Amazon Elastic Compute Cloud, Amazon Elastic Block Store, Amazon Simple Notification Service, Elastic Load Balancing, and Auto Scaling to build highly reliable, highly scalable, cost-effective applications without creating or configuring the underlying Amazon Web Services infrastructure.

With CloudFormation, you declare all your resources and dependencies in a template file. The template defines a collection of resources as a single unit called a stack. CloudFormation creates and deletes all member resources of the stack together and manages all dependencies between the resources for you.

For more information about CloudFormation, see the CloudFormation product page. CloudFormation makes use of other Amazon Web Services products. If you need additional technical information about a specific Amazon Web Services product, you can find the product's technical documentation at docs.aws.amazon.com.

### Usage

```r
cloudformation(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

### Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
– **anonymous**: Set anonymous credentials.

- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- cloudformation(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
  )
)
```
```python
sts_regional_endpoint = "string"
),
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
endpoint = "string",
region = "string"
)

## Operations

**activate_organizations_access**
Activate trusted access with Organizations

**activate_type**
Activates a public third-party extension, making it available for use in stack templates

**batch_describe_type_configurations**
Returns configuration data for the specified CloudFormation extensions, from the CloudFormation registry for the account and Region

**cancel_update_stack**
Cancels an update on the specified stack

**continue_update_rollback**
For a specified stack that’s in the UPDATE_ROLLBACK_FAILED state, continues rolling back the stack to the UPDATE_ROLLBACK_COMPLETE state

**create_change_set**
Creates a list of changes that will be applied to a stack so that you can review the changes before executing them

**create_generated_template**
Creates a template from existing resources that are not already managed with CloudFormation

**create_stack**
Creates a stack as specified in the template

**create_stack_instances**
Creates stack instances for the specified accounts, within the specified Amazon Web Services Regions

**create_stack_set**
Creates a stack set

**deactivate_organizations_access**
Deactivates trusted access with Organizations

**deactivate_type**
Deactivates a public extension that was previously activated in this account and Region

**delete_change_set**
Deletes the specified change set

**delete_generated_template**
Deletes a generated template

**delete_stack**
Deletes a stack

**delete_stack_instances**
Deletes stack instances for the specified accounts, in the specified Amazon Web Services Regions

**delete_stack_set**
Deletes a stack set

**deregister_type**
Marks an extension or extension version as DEPRECATED in the CloudFormation registry, removing it from active use

**describe_account_limits**
Retrieves your account’s CloudFormation limits, such as the maximum number of stacks that you can create in your account

**describe_change_set**
Returns the inputs for the change set and a list of changes that CloudFormation will make if you execute the change set

**describe_change_set_hooks**
Returns hook-related information for the change set and a list of changes that CloudFormation makes when you run the change set

**describe_generated_template**
Describes a generated template

**describe_organizations_access**
Retrieves information about the account’s OrganizationAccess status

**describe_publisher**
Returns information about a CloudFormation extension publisher

**describe_resource_scan**
Describes details of a resource scan

**describe_stack_drift_detection_status**
Returns information about a stack drift detection operation

**describe_stack_events**
Returns all stack related events for a specified stack in reverse chronological order

**describe_stack_instance**
Returns the stack instance that’s associated with the specified StackSet, Amazon Web Services account, and Amazon Web Services Region

**describe_stack_resource**
Returns a description of the specified resource in the specified stack

**describe_stack_resource_drifts**
Returns drift information for the resources that have been checked for drift in the specified stack

**describe_stack_resources**
Returns Amazon Web Services resource descriptions for running and deleted stacks

```
describe_stacks
describe_stack_set
describe_stack_set_operation
describe_type
describe_type_registration
detect_stack_drift
detect_stack_resource_drift
detect_stack_set_drift
estimate_template_cost
evaluate_change_set
generate_template
generate_template_summary
import_stacks_to_stack_set
list_change_sets
list_exports
list_generated_templates
list_imports
list_resource_scan_related_resources
list_resource_scan_resources
list_resource_scans
list_stack_instance_resource_drifts
list_stack_instances
list_stack_resources
list_stacks
list_stack_set_auto_deployment_targets
list_stack_set_operation_results
list_stack_sets
list_type_registrations
list_types
list_type_versions
publish_type
record_handler_progress
register_publisher
register_type
rollback_stack
set_stack_policy
set_type_configuration
set_type_default_version
signal_resource
start_resource_scan
stop_stack_set_operation
test_type
update_generated_template
update_stack
update_stack_instances

- Returns the description for the specified stack; if no stack name was specified, then it returns the description for all the stacks created
- Returns the description of the specified StackSet
- Returns the description of the specified StackSet operation
- Returns detailed information about an extension that has been registered
- Returns information about an extension’s registration, including its current status and type and version identifiers
- Detects whether a stack’s actual configuration differs, or has drifted, from its expected configuration, as defined in the stack template and any values specified as template parameters
- Returns information about whether a resource’s actual configuration differs, or has drifted, from its expected configuration, as defined in the stack template and any values specified as template parameters
- Detect drift on a stack set
- Returns the estimated monthly cost of a template
- Updates a stack using the input information that was provided when the specified change set was created
- Retrieves a generated template
- Returns the stack policy for a specified stack
- Returns the template body for a specified stack
- Returns information about a new or existing template
- Import existing stacks into a new stack sets
- Returns the ID and status of each active change set for a stack
- Lists all exported output values in the account and Region in which you call this action
- Lists your generated templates in this Region
- Lists all stacks that are importing an exported output value
- Lists the related resources for a list of resources from a resource scan
- Lists the resources from a resource scan
- List the resource scans from newest to oldest
- Returns drift information for resources in a stack instance
- Returns summary information about stack instances that are associated with the specified stack set
- Returns descriptions of all resources of the specified stack
- Returns the summary information for stacks whose status matches the specified StackStatusFilter
- Returns summary information about deployment targets for a stack set
- Returns summary information about the results of a stack set operation
- Returns summary information about operations performed on a stack set
- Returns summary information about stack sets that are associated with the user
- Returns a list of registration tokens for the specified extension(s)
- Returns summary information about extension that have been registered with CloudFormation
- Returns summary information about the versions of an extension
- Publishes the specified extension to the CloudFormation registry as a public extension
- Reports progress of a resource handler to CloudFormation
- Registers your account as a publisher of public extensions in the CloudFormation registry
- Registers an extension with the CloudFormation service
- When specifying RollbackStack, you preserve the state of previously provisioned resources
- Sets a stack policy for a specified stack
- Specifies the configuration data for a registered CloudFormation extension, in the given account and Region
- Specify the default version of an extension
- Sends a signal to the specified resource with a success or failure status
- Starts a scan of the resources in this account in this Region
- Stops an in-progress operation on a stack set and its associated stack instances
- Tests a registered extension to make sure it meets all necessary requirements for being used
- Updates a generated template
- Updates a stack as specified in the template
- Updates the parameter values for stack instances for the specified accounts, within the stack set
update_stack_set Updates the stack set, and associated stack instances in the specified accounts and Amazon Web Services Regions
update_termination_protection Updates termination protection for the specified stack
validate_template Validates a specified template

Examples

```r
## Not run:
svc <- cloudformation()
# This example creates a generated template with a resources file.
svc$create_generated_template(
  GeneratedTemplateName = "JazzyTemplate",
  Resources = list(
    list(
      ResourceIdentifier = list(
        BucketName = "jazz-bucket"
      ),
      ResourceType = "AWS::S3::Bucket"
    ),
    list(
      ResourceIdentifier = list(
        DhcpOptionsId = "random-id123"
      ),
      ResourceType = "AWS::EC2::DHCPOptions"
    )
  )
)
## End(Not run)
```

cloudfront

Amazon CloudFront

Description

This is the Amazon CloudFront API Reference. This guide is for developers who need detailed information about CloudFront API actions, data types, and errors. For detailed information about CloudFront features, see the Amazon CloudFront Developer Guide.

Usage

```r
cloudfront(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```
Arguments

config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends xmlns=

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

svc <- cloudfront(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string",
      )
    )
  )
)
secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
),
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

associate_alias
associate_alias Associates an alias (also known as a CNAME or an alternate domain name) with a CloudFront distribution

copy_distribution
create_cache_policy
create_cache_policy Creates a staging distribution using the configuration of the provided primary distribution

create_cloud_front_origin_access_identity
create_cloud_front_origin_access_identity Creates a new origin access identity

create_continuous_deployment_policy
create_continuous_deployment_policy Creates a continuous deployment policy that distributes traffic for a custom domain name to two different CloudFront distributions

create_distribution
create_distribution Create a CloudFront distribution

create_distribution_with_tags
create_distribution_with_tags Create a new distribution with tags

create_field_level_encryption_config
create_field_level_encryption_config Create a new field-level encryption configuration

create_field_level_encryption_profile
create_field_level_encryption_profile Create a field-level encryption profile

create_function
create_function Creates a CloudFront function

create_invalidation
create_invalidation Create a new invalidation

create_key_group
create_key_group Creates a key group that you can use with CloudFront signed URLs and signed cookies

create_key_value_store
create_key_value_store Specifies the key value store resource to add to your account

create_monitoring_subscription
create_monitoring_subscription Enables additional CloudWatch metrics for the specified CloudFront distribution

create_origin_access_control
create_origin_access_control Creates a new origin access control in CloudFront

create_origin_request_policy
create_origin_request_policy Creates an origin request policy

create_public_key
create_public_key Uploads a public key to CloudFront that you can use with signed URLs and signed cookies

create_realtime_log_config
create_realtime_log_config Creates a real-time log configuration

create_response_headers_policy
create_response_headers_policy Creates a response headers policy

create_streaming_distribution
create_streaming_distribution This API is deprecated
create_streaming_distribution_with_tags

delete_cache_policy

delete_cloud_front_origin_access_identity

delete_continuous_deployment_policy

delete_distribution

delete_field_level_encryption_config

delete_field_level_encryption_profile

delete_function

delete_key_group

delete_key_value_store

delete_monitoring_subscription

delete_origin_access_control

delete_origin_request_policy

delete_public_key

delete_realtime_log_config

delete_response_headers_policy

delete_streaming_distribution

describe_function

describe_key_value_store

describe_cache_policy

describe_cache_policy_config

describe_cloud_front_origin_access_identity

describe_cloud_front_origin_access_identity_config

describe_continuous_deployment_policy

describe_continuous_deployment_policy_config

describe_distribution

describe_distribution_config

describe_field_level_encryption

describe_field_level_encryption_config

describe_field_level_encryption_profile

describe_field_level_encryption_profile_config

describe_function

describe_invalidation

describe_key_group

describe_key_group_config

describe_monitoring_subscription

describe_origin_access_control

describe_origin_access_control_config

describe_origin_request_policy

describe_origin_request_policy_config

describe_public_key

describe_public_key_config

describe_realtime_log_config

describe_response_headers_policy

describe_response_headers_policy_config

describe_streaming_distribution

describe_streaming_distribution_config

list_cache_policies

This API is deprecated

Delete a cache policy

Delete an origin access identity

Delete a continuous deployment policy

Delete a distribution

Remove a field-level encryption configuration

Remove a field-level encryption profile

Deletes a CloudFront function

Deletes a key group

Specifies the key value store to delete

Disables additional CloudWatch metrics for the specified CloudFront distribution

Deletes a CloudFront origin access control

Deletes an origin request policy

Remove a public key you previously added to CloudFront

Deletes a real-time log configuration

Deletes a response headers policy

Delete a streaming distribution

Gets configuration information and metadata about a CloudFront function

Specifies the key value store and its configuration

Gets a cache policy, including the following metadata:

Gets a cache policy configuration

Get the information about an origin access identity

Get the configuration information about an origin access identity

Gets a continuous deployment policy, including metadata (the policy’s identifier and the date and time when the policy was last modified)

Gets configuration information about a continuous deployment policy

Get the information about a distribution

Get the configuration information about a distribution

Get the field-level encryption configuration information

Get the field-level encryption configuration information

Get the field-level encryption profile information

Get the field-level encryption profile configuration information

Gets the code of a CloudFront function

Get the information about an invalidation

Gets a key group, including the date and time when the key group was last modified

Gets a key group configuration

Gets information about whether additional CloudWatch metrics are enabled

Gets a CloudFront origin access control, including its unique identifier

Gets a CloudFront origin access control configuration

Gets an origin request policy, including the following metadata:

Gets an origin request policy configuration

Gets a public key

Gets a public key configuration

Gets a real-time log configuration

Gets a response headers policy, including metadata (the policy’s identifier and the date and time when the policy was last modified)

Gets a response headers policy configuration

Gets information about a specified RTMP distribution, including the distribution configuration

Get the configuration information about a streaming distribution

Gets a list of cache policies
list_cloud_front_origin_access_identities
list_conflicting_aliases
list_continuous_deployment_policies
list_distributions
list_distributions_by_cache_policy_id
list_distributions_by_key_group
list_distributions_by_origin_request_policy_id
list_distributions_by_response_headers_policy_id
list_distributions_by_web_acl_id
list_field_level_encryption_configs
list_field_level_encryption_profiles
list_functions
list_invalidations
list_key_groups
list_key_value_stores
list_origin_access_controls
list_origin_request_policies
list_public_keys
list_realtime_log_configs
list_response_headers_policies
list_streaming_distributions
list_tags_for_resource
publish_function
tag_resource
test_function
untag_resource
update_cache_policy
update_cloud_front_origin_access_identity
update_continuous_deployment_policy
update_distribution
update_distribution_with_staging_config
update_field_level_encryption_config
update_field_level_encryption_profile
update_function
update_key_group
update_key_value_store
update_origin_access_control
update_origin_request_policy
update_public_key
update_realtime_log_config
update_response_headers_policy
update_streaming_distribution

Lists origin access identities
Gets a list of aliases (also called CNAMEs or alternate domain names) that conflict or overlap with the provided alias, and the associated CloudFront distributions
List CloudFront distributions
Gets a list of distribution IDs for distributions that have a cache behavior associated with the specified cache policy
List distributions that have a cache behavior associated with the specified origin request policy
List the distributions that are associated with the specified WAF web ACL
List all field-level encryption configurations that have been created in CloudFront
Request a list of field-level encryption profiles that have been created in CloudFront
Gets a list of all CloudFront functions in your Amazon Web Services account
Lists invalidation batches
Gets a list of key groups
Specifies the key value stores to list
Gets the list of CloudFront origin access controls in this Amazon Web Services account
Gets a list of origin request policies
List all public keys that have been added to CloudFront for this account
List all response headers policies
List streaming distributions
List tags for a CloudFront resource
Publishes a CloudFront function by copying the function code from the DEVELOPMENT stage to the LIVE stage
Add tags to a CloudFront resource
Tests a CloudFront function
Remove tags from a CloudFront resource
Updates a cache policy configuration
Update an origin access identity
Updates a continuous deployment policy
Updates the configuration for a CloudFront distribution
Copies the staging distribution’s configuration to its corresponding primary distribution
Update a field-level encryption configuration
Update a field-level encryption profile
Updates a CloudFront function
Updates a key group
Specifies the key value store to update
Updates CloudFront origin access control
Updates an origin request policy configuration
Update public key information
Updates a real-time log configuration
Updates a response headers policy
Update a streaming distribution

Examples

## Not run:
cloudhsm <- cloudhsm()
# Use the following command to create a function.
svc$create_function(
    FunctionCode = "function-code.js",
    FunctionConfig = list(
        Comment = "my-function-comment",
        KeyValueStoreAssociations = list(
            Items = list(
                KeyValueStoreARN = "arn:aws:cloudfront::123456789012:key-value-st..."
            ),
            Quantity = 1L
        ),
        Runtime = "cloudfront-js-2.0"
    ),
    Name = "my-function-name"
)
## End(Not run)

---

### Description

AWS CloudHSM Service

This is documentation for AWS CloudHSM Classic. For more information, see AWS CloudHSM Classic FAQs, the AWS CloudHSM Classic User Guide, and the AWS CloudHSM Classic API Reference.

For information about the current version of AWS CloudHSM, see AWS CloudHSM, the AWS CloudHSM User Guide, and the AWS CloudHSM API Reference.

### Usage

```
cloudhsm(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

### Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      * **access_key_id**: AWS access key ID
      * **secret_access_key**: AWS secret access key
      * **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
– **anonymous**: Set anonymous credentials.

- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials**

Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint**

Optional shorthand for complete URL to use for the constructed client.

**region**

Optional shorthand for AWS Region used in instantiating the client.

## Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

### Service syntax

```
svc <- cloudhsm(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
  )
)
```
cloudhsm

```r

sts_region_endpoint = "string"
),
credentials = list(
  creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
  ),
  profile = "string",
  anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

```add_tags_to_resource``` This is documentation for AWS CloudHSM Classic
```create_hapg``` This is documentation for AWS CloudHSM Classic
```create_hsm``` This is documentation for AWS CloudHSM Classic
```create_luna_client``` This is documentation for AWS CloudHSM Classic
```delete_hapg``` This is documentation for AWS CloudHSM Classic
```delete_hsm``` This is documentation for AWS CloudHSM Classic
```delete_luna_client``` This is documentation for AWS CloudHSM Classic
```describe_hapg``` This is documentation for AWS CloudHSM Classic
```describe_hsm``` This is documentation for AWS CloudHSM Classic
```describe_luna_client``` This is documentation for AWS CloudHSM Classic
```get_config``` This is documentation for AWS CloudHSM Classic
```list_available_zones``` This is documentation for AWS CloudHSM Classic
```list_hapgs``` This is documentation for AWS CloudHSM Classic
```list_hsms``` This is documentation for AWS CloudHSM Classic
```list_luna_clients``` This is documentation for AWS CloudHSM Classic
```list_tags_for_resource``` This is documentation for AWS CloudHSM Classic
```modify_hapg``` This is documentation for AWS CloudHSM Classic
```modify_hsm``` This is documentation for AWS CloudHSM Classic
```modify_luna_client``` This is documentation for AWS CloudHSM Classic
```remove_tags_from_resource``` This is documentation for AWS CloudHSM Classic

Examples

```r
## Not run:
svc <- cloudhsm()
svc$add_tags_to_resource(
  Foo = 123
)
```

## End(Not run)
cloudhsmv2

AWS CloudHSM V2

Description

For more information about AWS CloudHSM, see AWS CloudHSM and the AWS CloudHSM User Guide.

Usage

cloudhsmv2(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
– **session_token**: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- cloudhsmv2(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string",
  close_connection = "logical",
  timeout = "numeric",
  s3_force_path_style = "logical",
  sts_regional_endpoint = "string"
),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

**Operations**
Amazon CloudSearch

Description

Amazon CloudSearch Configuration Service

You use the Amazon CloudSearch configuration service to create, configure, and manage search domains. Configuration service requests are submitted using the AWS Query protocol. AWS Query requests are HTTP or HTTPS requests submitted via HTTP GET or POST with a query parameter named Action.

The endpoint for configuration service requests is region-specific: cloudsearch.region.amazonaws.com. For example, cloudsearch.us-east-1.amazonaws.com. For a current list of supported regions and endpoints, see Regions and Endpoints.
Usage

cloudsearch(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

   config Optional configuration of credentials, endpoint, and/or region.
       • credentials:
           – creds:
               * access_key_id: AWS access key ID
               * secret_access_key: AWS secret access key
               * session_token: AWS temporary session token
           – profile: The name of a profile to use. If not given, then the default
                      profile is used.
           – anonymous: Set anonymous credentials.
       • endpoint: The complete URL to use for the constructed client.
       • region: The AWS Region used in instantiating the client.
       • close_connection: Immediately close all HTTP connections.
       • timeout: The time in seconds till a timeout exception is thrown when at-
                  tempting to make a connection. The default is 60 seconds.
       • s3_force_path_style: Set this to true to force the request to use path-style
                      addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
       • sts_regional_endpoint: Set sts regional endpoint resolver to regional or

   credentials Optional credentials shorthand for the config parameter
       • creds:
           – access_key_id: AWS access key ID
           – secret_access_key: AWS secret access key
           – session_token: AWS temporary session token
       • profile: The name of a profile to use. If not given, then the default profile
                  is used.
       • anonymous: Set anonymous credentials.

   endpoint Optional shorthand for complete URL to use for the constructed client.

   region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...),
where svc is the name you’ve assigned to the client. The available operations are listed in the Op-
erations section.
Service syntax

```r
svc <- cloudsearch(
  config = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string",
  close_connection = "logical",
  timeout = "numeric",
  s3_force_path_style = "logical",
  sts_regional_endpoint = "string"
),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `build_suggesters`: Indexes the search suggestions
- `create_domain`: Creates a new search domain
- `define_analysis_scheme`: Configures an analysis scheme that can be applied to a text or text-array field to define language-specific text processing options
- `define_expression`: Configures an Expression for the search domain
- `define_index_field`: Configures an IndexField for the search domain
- `define_suggester`: Configures a suggester for a domain
- `delete_analysis_scheme`: Deletes an analysis scheme
- `delete_domain`: Permanently deletes a search domain and all of its data
- `delete_expression`: Removes an Expression from the search domain
- `delete_index_field`: Removes an IndexField from the search domain
- `delete_suggester`: Deletes a suggester
- `describe_analysis_schemes`: Gets the analysis schemes configured for a domain
- `describe_availability_options`: Gets the availability options configured for a domain
- `describe_domain_endpoint_options`: Returns the domain’s endpoint options, specifically whether all requests to the domain must arrive over HTTPS
cloudsearchdomain

### Not run:
svc <- cloudsearch()
svc$build_suggesters(
  Foo = 123
)

### End(Not run)

cloudsearchdomain  Amazon CloudSearch Domain

**Description**

You use the AmazonCloudSearch2013 API to upload documents to a search domain and search those documents.

The endpoints for submitting `upload_documents`, `search`, and `suggest` requests are domain-specific. To get the endpoints for your domain, use the Amazon CloudSearch configuration service `DescribeDomains` action. The domain endpoints are also displayed on the domain dashboard in the Amazon CloudSearch console. You submit suggest requests to the search endpoint.

For more information, see the Amazon CloudSearch Developer Guide.

**Usage**

```r
cloudsearchdomain(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```
Arguments

config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-en.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- cloudsearchdomain(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
      
```


```r

secret_access_key = "string",
session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string",
close_connection = "logical",
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
  creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
  ),
  profile = "string",
  anonymous = "logical"
 ),
endpoint = "string",
region = "string"
)

Operations

- **search** Retrieves a list of documents that match the specified search criteria
- **suggest** Retrieves autocomplete suggestions for a partial query string
- **upload_documents** Posts a batch of documents to a search domain for indexing

Examples

```r

## Not run:
svc <- cloudsearchdomain()
svc$search(
  Foo = 123
)

## End(Not run)
```
Description

CloudTrail

This is the CloudTrail API Reference. It provides descriptions of actions, data types, common parameters, and common errors for CloudTrail.

CloudTrail is a web service that records Amazon Web Services API calls for your Amazon Web Services account and delivers log files to an Amazon S3 bucket. The recorded information includes the identity of the user, the start time of the Amazon Web Services API call, the source IP address, the request parameters, and the response elements returned by the service.

As an alternative to the API, you can use one of the Amazon Web Services SDKs, which consist of libraries and sample code for various programming languages and platforms (Java, Ruby, .NET, iOS, Android, etc.). The SDKs provide programmatic access to CloudTrail. For example, the SDKs handle cryptographically signing requests, managing errors, and retrying requests automatically. For more information about the Amazon Web Services SDKs, including how to download and install them, see Tools to Build on Amazon Web Services.

See the CloudTrail User Guide for information about the data that is included with each Amazon Web Services API call listed in the log files.

Usage

```r
cloudtrail(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- `config`: Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
    - `creds`:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
    - `anonymous`: Set anonymous credentials.
  - `endpoint`: The complete URL to use for the constructed client.
  - `region`: The AWS Region used in instantiating the client.
  - `close_connection`: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

• **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.

• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials**  
Optional credentials shorthand for the config parameter

• **creds**:
  – **access_key_id**: AWS access key ID
  – **secret_access_key**: AWS secret access key
  – **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

**endpoint**  
Optional shorthand for complete URL to use for the constructed client.

**region**  
Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
csvc <- cloudtrail(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  )
)
```
secret_access_key = "string",
session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
);

Operations

add_tags  Adds one or more tags to a trail, event data store, or channel, up to a limit of 50
cancel_query  Cancels a query if the query is not in a terminated state, such as CANCELLED, FAILED, TIMED_OUT, or FINISHED
create_channel  Creates a channel for CloudTrail to ingest events from a partner or external source
create_event_data_store  Creates a new event data store
create_trail  Creates a trail that specifies the settings for delivery of log data to an Amazon S3 bucket
delete_channel  Deletes a channel
delete_event_data_store  Disables the event data store specified by EventDataStore, which accepts an event data store ARN
delete_resource_policy  Deletes the resource-based policy attached to the CloudTrail channel
delete_trail  Deletes a trail
deregister_organization_delegated_admin  Removes CloudTrail delegated administrator permissions from a member account in an organization
describe_query  Returns metadata about a query, including query run time in milliseconds, number of events scanned and matched, and query status
describe_trails  Retrieves settings for one or more trails associated with the current Region for your account
disable_federation  Disables Lake query federation on the specified event data store
enable_federation  Enables Lake query federation on the specified event data store
get_channel  Returns information about a specific channel
get_event_data_store  Returns information about an event data store specified as either an ARN or the ID portion of the ARN
get_event_selectors  Describes the settings for the event selectors that you configured for your trail
get_insight_selectors  Describes the settings for the Insights event selectors that you configured for your trail
get_import  Returns information about a specific import
get_query_results  Gets event data results of a query
get_resource_policy  Returns the JSON text of the resource-based policy document attached to the CloudTrail channel
get_trail  Returns settings information for a specified trail
get_trail_status  Returns a JSON-formatted list of information about the specified trail
get_channels  Lists the channels in the current account, and their source names
get_event_data_stores  Returns information about all event data stores in the account, in the current Region
get_trail_status  Returns a list of failures for the specified import
list_channels  Returns information on all imports, or a select set of imports by ImportStatus or Destination
list_event_data_stores  Returns Insights metrics data for trails that have enabled Insights
list_import_failures  Returns all public keys whose private keys were used to sign the digest files within a specified time range
list_imports  Returns a list of queries and query statuses for the past seven days
list_insights_metric_data  Lists the tags for the specified trails, event data stores, or channels in the current Region
list_public_keys  Lists trails that are in the current account
list_queries  Looks up management events or CloudTrail Insights events that are captured by CloudTrail
lookup_events  Configures an event selector or advanced event selectors for your trail
put_event_selectors  Lets you enable Insights event logging by specifying the Insights selectors that you want to enable
pull_tags  Adds a resource-based permission policy to a CloudTrail channel that is used for the import
put_insight_selectors
put_resource_policy
**cloudtraildataservice**

### Description

The CloudTrail Data Service lets you ingest events into CloudTrail from any source in your hybrid environments, such as in-house or SaaS applications hosted on-premises or in the cloud, virtual machines, or containers. You can store, access, analyze, troubleshoot and take action on this data without maintaining multiple log aggregators and reporting tools. After you run `put_audit_events` to ingest your application activity into CloudTrail, you can use CloudTrail Lake to search, query, and analyze the data that is logged from your applications.

### Usage

```r
cloudtraildataservice(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

### Examples

```r
## Not run:
svc <- cloudtrail()
svc$add_tags(
  Foo = 123
)
## End(Not run)
```

Registers an organization’s member account as the CloudTrail delegated administrator

Removes the specified tags from a trail, event data store, or channel

Restores a deleted event data store specified by EventDataStore, which accepts an event data store ARN

Starts the ingestion of live events on an event data store specified as either an ARN or the ID portion of the ARN

Starts an import of logged trail events from a source S3 bucket to a destination event data store

Starts the recording of Amazon Web Services API calls and log file delivery for a specified trail

Stops the ingestion of live events on an event data store specified as either an ARN or the ID portion of the ARN

Stops a specified import

Suspends the recording of Amazon Web Services API calls and log file delivery for a specified trail

Updates a channel specified by a required channel ARN or UUID

Updates an event data store

Updates trail settings that control what events you are logging, and how to handle log files
Arguments

config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    - creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    - profile: The name of a profile to use. If not given, then the default profile is used.
    - anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    - access_key_id: AWS access key ID
    - secret_access_key: AWS secret access key
    - session_token: AWS temporary session token
    • profile: The name of a profile to use. If not given, then the default profile is used.
    • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```
svc <- cloudtraildataservice(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string",
        profile = "string",
        anonymous = FALSE
      )
    ),
    endpoint = "string",
    region = "string",
    close_connection = FALSE,
    timeout = 60,
    s3_force_path_style = FALSE,
    sts_regional_endpoint = "string"
  )
)
```
Operations

`put_audit_events`  Ingests your application events into CloudTrail Lake

Examples

```r
## Not run:
svc <- cloudtraildataservice()
svc$put_audit_events(
  Foo = 123
)

## End(Not run)
```

---

`cloudwatch`  
*Amazon CloudWatch*
Description

Amazon CloudWatch monitors your Amazon Web Services (Amazon Web Services) resources and the applications you run on Amazon Web Services in real time. You can use CloudWatch to collect and track metrics, which are the variables you want to measure for your resources and applications. CloudWatch alarms send notifications or automatically change the resources you are monitoring based on rules that you define. For example, you can monitor the CPU usage and disk reads and writes of your Amazon EC2 instances. Then, use this data to determine whether you should launch additional instances to handle increased load. You can also use this data to stop under-used instances to save money.

In addition to monitoring the built-in metrics that come with Amazon Web Services, you can monitor your own custom metrics. With CloudWatch, you gain system-wide visibility into resource utilization, application performance, and operational health.

Usage

cloudwatch(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

class config Optional configuration of credentials, endpoint, and/or region.

    • credentials:
      – creds:
        * access_key_id: AWS access key ID
        * secret_access_key: AWS secret access key
        * session_token: AWS temporary session token
      – profile: The name of a profile to use. If not given, then the default profile is used.
      – anonymous: Set anonymous credentials.

    • endpoint: The complete URL to use for the constructed client.

    • region: The AWS Region used in instantiating the client.

    • close_connection: Immediately close all HTTP connections.

    • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

    • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

    • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ent.html

class credentials Optional credentials shorthand for the config parameter

    • creds:
- `access_key_id`: AWS access key ID
- `secret_access_key`: AWS secret access key
- `session_token`: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

`endpoint`  Optional shorthand for complete URL to use for the constructed client.

`region`  Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- cloudwatch(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```
Operations

```r
delete_alarms          Deletes the specified alarms
delete_anomaly_detector Deletes the specified anomaly detection model from your account
delete_dashboards      Deletes all dashboards that you specify
delete_insight_rules   Permanently deletes the specified Contributor Insights rules
delete_metric_stream   Permanently deletes the metric stream that you specify
describe_alarm_history Retrieves the history for the specified alarm
describe_alarms         Retrieves the specified alarms
describe_alarms_for_metric Lists the anomaly detection models that you have created in your account
describe_anomaly_detectors Returns a list of all the Contributor Insights rules in your account
disable_alarm_actions  Disables the actions for the specified alarms
disable_insight_rules  Disables the specified Contributor Insights rules
enable_alarm_actions    Enables the actions for the specified alarms
enable_insight_rules    Enables the specified Contributor Insights rules
get_dashboard          Displays the details of the dashboard that you specify
get_insight_rule_report This operation returns the time series data collected by a Contributor Insights rule
get_metric_data        You can use the GetMetricData API to retrieve CloudWatch metric values
get_metric_statistics  Gets statistics for the specified metric
get_metric_stream      Returns information about the metric stream that you specify
get_metric_widget_image You can use the GetMetricWidgetImage API to retrieve a snapshot graph of one or more Amazon CloudWatch metrics
list_dashboards        Returns a list of the dashboards for your account
list_managed_insight_rules Returns a list that contains the number of managed Contributor Insights rules in your account
list_metrics           List the specified metrics
list_metric_streams    Returns a list of metric streams in this account
list_tags_for_resource Displays the tags associated with a CloudWatch resource
put_anomaly_detector   Creates an anomaly detection model for a CloudWatch metric
put_composite_alarm    Creates or updates a composite alarm
put_dashboard          Creates a dashboard if it does not already exist, or updates an existing dashboard
put_insight_rule       Creates a Contributor Insights rule
put Managed_insight_rules Creates a managed Contributor Insights rule for a specified Amazon Web Services resource
put_metric_alarm       Creates or updates an alarm and associates it with the specified metric, metric math expression,
put_metric_data        Publishes metric data points to Amazon CloudWatch
put_metric_stream      Creates or updates a metric stream
set_alarm_state        Temporarily sets the state of an alarm for testing purposes
start_metric_streams   Starts the streaming of metrics for one or more of your metric streams
stop_metric_streams    Stops the streaming of metrics for one or more of your metric streams
tag_resource          Assigns one or more tags (key-value pairs) to the specified CloudWatch resource
untag_resource         Removes one or more tags from the specified resource
```

Examples

```r
## Not run:
svc <- cloudwatch()
svc$delete_alarms(
  Foo = 123
)```
Amazon CloudWatch Events

Description

Amazon EventBridge helps you to respond to state changes in your Amazon Web Services resources. When your resources change state, they automatically send events to an event stream. You can create rules that match selected events in the stream and route them to targets to take action. You can also use rules to take action on a predetermined schedule. For example, you can configure rules to:

- Automatically invoke an Lambda function to update DNS entries when an event notifies you that Amazon EC2 instance enters the running state.
- Direct specific API records from CloudTrail to an Amazon Kinesis data stream for detailed analysis of potential security or availability risks.
- Periodically invoke a built-in target to create a snapshot of an Amazon EBS volume.

For more information about the features of Amazon EventBridge, see the Amazon EventBridge User Guide.

Usage

cloudwatchevents(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
• **region**: The AWS Region used in instantiating the client.
• **close_connection**: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter

• **creds**:  
  – **access_key_id**: AWS access key ID  
  – **secret_access_key**: AWS secret access key  
  – **session_token**: AWS temporary session token
• **profile**: The name of a profile to use. If not given, then the default profile is used.
• **anonymous**: Set anonymous credentials.

event Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- cloudwatchevents(  
  config = list(  
    credentials = list(  
      creds = list(  
        access_key_id = "string",  
        secret_access_key = "string",  
        session_token = "string"  
      ),  
      profile = "string",  
      anonymous = "logical"  
    ),  
    endpoint = "string",  
    region = "string",  
    close_connection = "logical",  
    timeout = "numeric",  
    s3_force_path_style = "logical",  
    sts_regional_endpoint = "string"  
  ),
)  
```
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations
activate_event_source  Activates a partner event source that has been deactivated
cancel_replay  Cancels the specified replay
create_api_destination  Creates an API destination, which is an HTTP invocation endpoint configured as a target for events
create_archive  Creates an archive of events with the specified settings
create_connection  Creates a connection
create_event_bus  Creates a new event bus within your account
deactivate_event_source  You can use this operation to temporarily stop receiving events from the specified partner event source
deauthorize_connection  Removes all authorization parameters from the connection
delete_api_destination  Deletes the specified API destination
delete_archive  Deletes the specified archive
delete_connection  Deletes a connection
delete_event_bus  Deletes the specified custom event bus or partner event bus
delete_partner_event_source  This operation is used by SaaS partners to delete a partner event source
delete_rule  Deletes the specified rule
describe_api_destination  Retrieves details about an API destination
describe_archive  Retrieves details about an archive
describe_connection  Retrieves details about a connection
describe_event_bus  Displays details about an event bus in your account
describe_event_source  This operation lists details about a partner event source that is shared with your account
describe_partner_event_source  An SaaS partner can use this operation to list details about a partner event source that their integration is sharing
describe_replay  Retrieves details about a replay
describe_rule  Describes the specified rule
disable_rule  Disables the specified rule
enable_rule  Enables the specified rule
list_api_destinations  Retrieves a list of API destination in the account in the current Region
list_archives  Lists your archives
list_connections  Retrieves a list of connections from the account
list_event_buses  Lists all the event buses in your account, including the default event bus, custom event buses, and partner event buses
list_event_sources  You can use this to see all the partner event sources that have been shared with your Amazon Web Services account
list_partner_event_source_accounts  An SaaS partner can use this operation to display the Amazon Web Services account ID that a particular partner event source name is associated with
list_partner_event_sources  An SaaS partner can use this operation to list all the partner event source names that they have created
list_replays  Lists your replays
list_rule_names_by_target: Lists the rules for the specified target
list_rules: Lists your Amazon EventBridge rules
list_tags_for_resource: Displays the tags associated with an EventBridge resource
list_targets_by_rule: Lists the targets assigned to the specified rule
put_events: Sends custom events to Amazon EventBridge so that they can be matched to rules
put_partner_events: This is used by SaaS partners to write events to a customer’s partner event bus
put_permission: Running PutPermission permits the specified Amazon Web Services account or Amazon Web Services organization to put events to the specified event bus
put_rule: Creates or updates the specified rule
put_targets: Adds the specified targets to the specified rule, or updates the targets if they are already assigned
remove_permission: Revokes the permission of another Amazon Web Services account to be able to put events to the specified event bus
remove_targets: Removes the specified targets from the specified rule
start_replay: Starts the specified replay
tag_resource: Assigns one or more tags (key-value pairs) to the specified EventBridge resource
test_event_pattern: Tests whether the specified event pattern matches the provided event
untag_resource: Removes one or more tags from the specified EventBridge resource
update_api_destination: Updates an API destination
update_archive: Updates the specified archive
update_connection: Updates settings for a connection

Examples

```r
## Not run:
svc <- cloudwatchevidently()
svc$activate_event_source(
  Foo = 123
)

## End(Not run)
```

cloudwatchevidently Amazon CloudWatch Evidently

**Description**

You can use Amazon CloudWatch Evidently to safely validate new features by serving them to a specified percentage of your users while you roll out the feature. You can monitor the performance of the new feature to help you decide when to ramp up traffic to your users. This helps you reduce risk and identify unintended consequences before you fully launch the feature.

You can also conduct A/B experiments to make feature design decisions based on evidence and data. An experiment can test as many as five variations at once. Evidently collects experiment data and analyzes it using statistical methods. It also provides clear recommendations about which variations perform better. You can test both user-facing features and backend features.
Usage

cloudwatchevidently(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

    config  Optional configuration of credentials, endpoint, and/or region.
        • credentials:
            – creds:
                * access_key_id: AWS access key ID
                * secret_access_key: AWS secret access key
                * session_token: AWS temporary session token
            – profile: The name of a profile to use. If not given, then the default profile is used.
            – anonymous: Set anonymous credentials.
        • endpoint: The complete URL to use for the constructed client.
        • region: The AWS Region used in instantiating the client.
        • close_connection: Immediately close all HTTP connections.
        • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
        • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
        • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

    credentials Optional credentials shorthand for the config parameter
        • creds:
            – access_key_id: AWS access key ID
            – secret_access_key: AWS secret access key
            – session_token: AWS temporary session token
        • profile: The name of a profile to use. If not given, then the default profile is used.
        • anonymous: Set anonymous credentials.

    endpoint  Optional shorthand for complete URL to use for the constructed client.

    region   Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```r
svc <- cloudwatchevidently(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- **batch_evaluate_feature**
  - This operation assigns feature variation to user sessions
- **create_experiment**
  - Creates an Evidently experiment
- **create_feature**
  - Creates an Evidently feature that you want to launch or test
- **create_launch**
  - Creates a launch of a given feature
- **create_project**
  - Creates a project, which is the logical object in Evidently that can contain features, launches, and experiments
- **create_segment**
  - Use this operation to define a segment of your audience
- **delete_experiment**
  - Deletes an Evidently experiment
- **delete_feature**
  - Deletes an Evidently feature
- **delete_launch**
  - Deletes an Evidently launch
- **delete_project**
  - Deletes an Evidently project
- **delete_segment**
  - Deletes a segment
- **evaluate_feature**
  - This operation assigns a feature variation to one given user session
- **get_experiment**
  - Returns the details about one experiment
- **get_experiment_results**
  - Retrieves the results of a running or completed experiment
### Examples

```r
## Not run:
svc <- cloudwatchevidently()
svc$batch_evaluate_feature(
  Foo = 123
)
## End(Not run)
```

---

### cloudwatchinternetmonitor

**Amazon CloudWatch Internet Monitor**

### Description

Amazon CloudWatch Internet Monitor provides visibility into how internet issues impact the performance and availability between your applications hosted on Amazon Web Services and your end users. It can reduce the time it takes for you to diagnose internet issues from days to minutes.
Internet Monitor uses the connectivity data that Amazon Web Services captures from its global networking footprint to calculate a baseline of performance and availability for internet traffic. This is the same data that Amazon Web Services uses to monitor internet uptime and availability. With those measurements as a baseline, Internet Monitor raises awareness for you when there are significant problems for your end users in the different geographic locations where your application runs.

Internet Monitor publishes internet measurements to CloudWatch Logs and CloudWatch Metrics, to easily support using CloudWatch tools with health information for geographies and networks specific to your application. Internet Monitor sends health events to Amazon EventBridge so that you can set up notifications. If an issue is caused by the Amazon Web Services network, you also automatically receive an Amazon Web Services Health Dashboard notification with the steps that Amazon Web Services is taking to mitigate the problem.

To use Internet Monitor, you create a monitor and associate your application’s resources with it - VPCs, NLBs, CloudFront distributions, or WorkSpaces directories - so Internet Monitor can determine where your application’s internet traffic is. Internet Monitor then provides internet measurements from Amazon Web Services that are specific to the locations and ASNs (typically, internet service providers or ISPs) that communicate with your application.

For more information, see Using Amazon CloudWatch Internet Monitor in the Amazon CloudWatch User Guide.

Usage

```
cloudwatchinternetmonitor(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      * **access_key_id**: AWS access key ID
      * **secret_access_key**: AWS secret access key
      * **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

**credentials**
Optional credentials shorthand for the config parameter

• **creds**:
  – **access_key_id**: AWS access key ID
  – **secret_access_key**: AWS secret access key
  – **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**
A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- cloudwatchinternetmonitor(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    )
  )
)
```
Operations

create_monitor Creates a monitor in Amazon CloudWatch Internet Monitor
delete_monitor Deletes a monitor in Amazon CloudWatch Internet Monitor
get_health_event Gets information that Amazon CloudWatch Internet Monitor has created and stored about a health event
get_internet_event Gets information that Amazon CloudWatch Internet Monitor has generated about an internet event
get_monitor Gets information about a monitor in Amazon CloudWatch Internet Monitor based on a monitor name
get_query_results Return the data for a query with the Amazon CloudWatch Internet Monitor query interface
get_query_status Returns the current status of a query for the Amazon CloudWatch Internet Monitor query interface, for a specified query ID and monitor
list_health_events Lists all health events for a monitor in Amazon CloudWatch Internet Monitor
list_internet_events Lists internet events that cause performance or availability issues for client locations
list_monitors Lists all of your monitors for Amazon CloudWatch Internet Monitor and their statuses, along with the Amazon Resource Name (ARN) and name of each monitor
list_tags_for_resource Lists the tags for a resource
start_query Start a query to return data for a specific query type for the Amazon CloudWatch Internet Monitor query interface
stop_query Stop a query that is progress for a specific monitor
tag_resource Adds a tag to a resource
untag_resource Removes a tag from a resource
update_monitor Updates a monitor

Examples

## Not run:
svc <- cloudwatchinternetmonitor()
svc$create_monitor(
  Foo = 123
)

## End(Not run)
Description

You can use Amazon CloudWatch Logs to monitor, store, and access your log files from EC2 instances, CloudTrail, and other sources. You can then retrieve the associated log data from CloudWatch Logs using the CloudWatch console. Alternatively, you can use CloudWatch Logs commands in the Amazon Web Services CLI, CloudWatch Logs API, or CloudWatch Logs SDK.

You can use CloudWatch Logs to:

- **Monitor logs from EC2 instances in real time**: You can use CloudWatch Logs to monitor applications and systems using log data. For example, CloudWatch Logs can track the number of errors that occur in your application logs. Then, it can send you a notification whenever the rate of errors exceeds a threshold that you specify. CloudWatch Logs uses your log data for monitoring so no code changes are required. For example, you can monitor application logs for specific literal terms (such as "NullReferenceException"). You can also count the number of occurrences of a literal term at a particular position in log data (such as "404" status codes in an Apache access log). When the term you are searching for is found, CloudWatch Logs reports the data to a CloudWatch metric that you specify.

- **Monitor CloudTrail logged events**: You can create alarms in CloudWatch and receive notifications of particular API activity as captured by CloudTrail. You can use the notification to perform troubleshooting.

- **Archive log data**: You can use CloudWatch Logs to store your log data in highly durable storage. You can change the log retention setting so that any log events earlier than this setting are automatically deleted. The CloudWatch Logs agent helps to quickly send both rotated and non-rotated log data off of a host and into the log service. You can then access the raw log data when you need it.

Usage

```python
cloudwatchlogs(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
• **region**: The AWS Region used in instantiating the client.
• **close_connection**: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

credentials Optional credentials shorthand for the config parameter

• **creds**:  
  – `access_key_id`: AWS access key ID  
  – `secret_access_key`: AWS secret access key  
  – `session_token`: AWS temporary session token
• **profile**: The name of a profile to use. If not given, then the default profile is used.
• **anonymous**: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service's operations using syntax like `svc$operation(...)`, where `svc` is the name you've assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- cloudwatchlogs(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
)```
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

associate_kms_key
cancel_export_task
create_delivery
create_export_task
create_log_anomaly_detector
create_log_group
create_log_stream
delete_account_policy
delete_data_protection_policy
delete_delivery
delete_delivery_destination
delete_delivery_destination_policy
delete_delivery_source
delete_destination
delete_log_anomaly_detector
delete_log_group
delete_log_stream
delete_metric_filter
delete_query_definition
delete_resource_policy
delete_retention_policy
delete_subscription_filter
describe_account_policies
describe_deliveries
describe_delivery_destinations
describe_delivery_sources
describe_destinations
describe_export_tasks
describe_log_groups
describe_log_streams
describe_metric_filters
describe_queries
describe_query_definitions

Associates the specified KMS key with either one log group in the account, or with all stored CloudWatch Logs query insights results in the account
Cancels the specified export task
Creates a delivery
Creates an export task so that you can efficiently export data from a log group to an Amazon S3 bucket
Creates an anomaly detector that regularly scans one or more log groups and look for patterns and anomalies in the logs
Creates a log group with the specified name
Creates a log stream for the specified log group
Deletes a CloudWatch Logs account policy
Deletes the data protection policy from the specified log group
Deletes a delivery destination
Deletes a delivery destination policy
Deletes a delivery source
Deletes the specified destination, and eventually disables all the subscription filters that publish to it
Deletes the specified CloudWatch Logs anomaly detector
Deletes the specified log group and permanently deletes all the archived log events associated with the log group
Deletes the specified log stream and permanently deletes all the archived log events associated with the log group
Deletes the specified metric filter
Deletes a saved CloudWatch Logs Insights query definition
Deletes a resource policy from this account
Deletes the specified retention policy
Deletes the specified subscription filter
Returns a list of all CloudWatch Logs account policies in the account
Retrieves a list of the deliveries that have been created in the account
Retrieves a list of the delivery destinations that have been created in the account
Lists all your destinations
Lists the specified export tasks
Lists the specified log groups
Lists the log streams for the specified log group
Lists the specified metric filters
Returns a list of CloudWatch Logs Insights queries that are scheduled, running, or have been run recently in this account
This operation returns a paginated list of your saved CloudWatch Logs Insights query definitions
### Examples

```r
## Not run:
svc <- cloudwatchlogs()
svc$associate_kms_key()
```
Use Amazon CloudWatch Observability Access Manager to create and manage links between source accounts and monitoring accounts by using CloudWatch cross-account observability. With CloudWatch cross-account observability, you can monitor and troubleshoot applications that span multiple accounts within a Region. Seamlessly search, visualize, and analyze your metrics, logs, traces, and Application Insights applications in any of the linked accounts without account boundaries.

Set up one or more Amazon Web Services accounts as monitoring accounts and link them with multiple source accounts. A monitoring account is a central Amazon Web Services account that can view and interact with observability data generated from source accounts. A source account is an individual Amazon Web Services account that generates observability data for the resources that reside in it. Source accounts share their observability data with the monitoring account. The shared observability data can include metrics in Amazon CloudWatch, logs in Amazon CloudWatch Logs, traces in X-Ray, and applications in Amazon CloudWatch Application Insights.

Usage

```r
cloudwatchobservabilityaccessmanager(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
      - **profile**: The name of a profile to use. If not given, then the default profile is used.
      - **anonymous**: Set anonymous credentials.
• **endpoint**: The complete URL to use for the constructed client.
• **region**: The AWS Region used in instantiating the client.
• **close_connection**: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endsWith.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endsWith.html)

**credentials**

Optional credentials shorthand for the config parameter

• **creds**:
  • access_key_id: AWS access key ID
  • secret_access_key: AWS secret access key
  • session_token: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

**endpoint**

Optional shorthand for complete URL to use for the constructed client.

**region**

Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- cloudwatchobservabilityaccessmanager(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  )
)
```
cloudwatchobservabilityaccessmanager

credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

create_link Creates a link between a source account and a sink that you have created in a monitoring account
create_sink Use this to create a sink in the current account, so that it can be used as a monitoring account in CloudWatch cross-account observability
delete_link Deletes a link between a monitoring account sink and a source account
delete_sink Deletes a sink
get_link Returns complete information about one link
get_sink Returns complete information about one monitoring account sink
get_sink_policy Returns the current sink policy attached to this sink
list_attached_links Returns a list of source account links that are linked to this monitoring account sink
list_links Use this operation in a source account to return a list of links to monitoring account sinks that this source account has
list_sinks Use this operation in a monitoring account to return the list of sinks created in that account
list_tags_for_resource Displays the tags associated with a resource
put_sink_policy Creates or updates the resource policy that grants permissions to source accounts to link to the monitoring account sink
tag_resource Assigns one or more tags (key-value pairs) to the specified resource
untag_resource Removes one or more tags from the specified resource
update_link Use this operation to change what types of data are shared from a source account to its linked monitoring account sink

Examples

## Not run:
svc <- cloudwatchobservabilityaccessmanager()
svc$create_link(
    Foo = 123
)

## End(Not run)
With Amazon CloudWatch RUM, you can perform real-user monitoring to collect client-side data about your web application performance from actual user sessions in real time. The data collected includes page load times, client-side errors, and user behavior. When you view this data, you can see it all aggregated together and also see breakdowns by the browsers and devices that your customers use.

You can use the collected data to quickly identify and debug client-side performance issues. CloudWatch RUM helps you visualize anomalies in your application performance and find relevant debugging data such as error messages, stack traces, and user sessions. You can also use RUM to understand the range of end-user impact including the number of users, geolocations, and browsers used.

Usage

```r
cloudwatchrum(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`. 

• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials**

Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.

- **anonymous**: Set anonymous credentials.

**endpoint**

Optional shorthand for complete URL to use for the constructed client.

**region**

Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- cloudwatchrum(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
```
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

batch_create_rum_metric_definitions
batch_delete_rum_metric_definitions
batch_get_rum_metric_definitions
create_app_monitor
delete_app_monitor
delete_rum_metrics_destination
get_app_monitor
get_app_monitor_data
list_app_monitors
list_rum_metrics_destinations
list_tags_for_resource
put_rum_events
put_rum_metrics_destination
tag_resource
untag_resource
update_app_monitor
update_rum_metric_definition

Specifications the extended metrics and custom metrics that you want a CloudWatch RUM app monitor to send to a destination.
Removes the specified metrics from being sent to an extended metrics destination.
Retrieves the list of metrics and dimensions that a RUM app monitor is sending to a single destination.
Creates a Amazon CloudWatch RUM app monitor, which collects telemetry data from your web application.
Deletes an existing app monitor.
Deletes a destination for CloudWatch RUM extended metrics, so that the specified app monitor stops sending extended metrics to that destination.
Retrieves the complete configuration information for one app monitor.
Retrieves the raw performance events that RUM has collected from your web application.
Returns a list of the Amazon CloudWatch RUM app monitors in the account.
Returns a list of destinations that you have created to receive RUM extended metrics, for the specified app monitor.
Displays the tags associated with a CloudWatch RUM resource.
Sends telemetry events about your application performance and user behavior to CloudWatch RUM.
Creates or updates a destination to receive extended metrics from CloudWatch RUM.
Assigns one or more tags (key-value pairs) to the specified CloudWatch RUM resource.
Removes one or more tags from the specified resource.
Updates the configuration of an existing app monitor.
Modifies one existing metric definition for CloudWatch RUM extended metrics.

Examples

```r
## Not run:
svc <- cloudwatchrum()
svc$batch_create_rum_metric_definitions(
  Foo = 123
)
## End(Not run)
```

Description

CodeArtifact is a fully managed artifact repository compatible with language-native package managers and build tools such as npm, Apache Maven, pip, and dotnet. You can use CodeArtifact to share packages with development teams and pull packages. Packages can be pulled from both public
and CodeArtifact repositories. You can also create an upstream relationship between a CodeArtifact repository and another repository, which effectively merges their contents from the point of view of a package manager client.

**CodeArtifact concepts**

- **Repository**: A CodeArtifact repository contains a set of package versions, each of which maps to a set of assets, or files. Repositories are polyglot, so a single repository can contain packages of any supported type. Each repository exposes endpoints for fetching and publishing packages using tools such as the npm CLI or the Maven CLI (mvn). For a list of supported package managers, see the CodeArtifact User Guide.

- **Domain**: Repositories are aggregated into a higher-level entity known as a domain. All package assets and metadata are stored in the domain, but are consumed through repositories. A given package asset, such as a Maven JAR file, is stored once per domain, no matter how many repositories it’s present in. All of the assets and metadata in a domain are encrypted with the same customer master key (CMK) stored in Key Management Service (KMS).

Each repository is a member of a single domain and can’t be moved to a different domain. The domain allows organizational policy to be applied across multiple repositories, such as which accounts can access repositories in the domain, and which public repositories can be used as sources of packages.

Although an organization can have multiple domains, we recommend a single production domain that contains all published artifacts so that teams can find and share packages across their organization.

- **Package**: A package is a bundle of software and the metadata required to resolve dependencies and install the software. CodeArtifact supports npm, PyPI, Maven, NuGet, Swift, Ruby, and generic package formats. For more information about the supported package formats and how to use CodeArtifact with them, see the CodeArtifact User Guide.

  In CodeArtifact, a package consists of:
  - A name (for example, webpack is the name of a popular npm package)
  - An optional namespace (for example, @types in @types/node)
  - A set of versions (for example, 1.0.0, 1.0.1, 1.0.2, etc.)
  - Package-level metadata (for example, npm tags)

- **Package group**: A group of packages that match a specified definition. Package groups can be used to apply configuration to multiple packages that match a defined pattern using package format, package namespace, and package name. You can use package groups to more conveniently configure package origin controls for multiple packages. Package origin controls are used to block or allow ingestion or publishing of new package versions, which protects users from malicious actions known as dependency substitution attacks.

- **Package version**: A version of a package, such as @types/node 12.6.9. The version number format and semantics vary for different package formats. For example, npm package versions must conform to the Semantic Versioning specification. In CodeArtifact, a package version consists of the version identifier, metadata at the package version level, and a set of assets.

- **Upstream repository**: One repository is upstream of another when the package versions in it can be accessed from the repository endpoint of the downstream repository, effectively merging the contents of the two repositories from the point of view of a client. CodeArtifact allows creating an upstream relationship between two repositories.
• **Asset**: An individual file stored in CodeArtifact associated with a package version, such as an npm `.tgz` file or Maven POM and JAR files.

**CodeArtifact supported API operations**

• `associate_external_connection`: Adds an existing external connection to a repository.
• `copy_package_versions`: Copies package versions from one repository to another repository in the same domain.
• `create_domain`: Creates a domain.
• `create_package_group`: Creates a package group.
• `create_repository`: Creates a CodeArtifact repository in a domain.
• `delete_domain`: Deletes a domain. You cannot delete a domain that contains repositories.
• `delete_domain_permissions_policy`: Deletes the resource policy that is set on a domain.
• `delete_package`: Deletes a package and all associated package versions.
• `delete_package_group`: Deletes a package group. Does not delete packages or package versions that are associated with a package group.
• `delete_package_versions`: Deletes versions of a package. After a package has been deleted, it can be republished, but its assets and metadata cannot be restored because they have been permanently removed from storage.
• `delete_repository`: Deletes a repository.
• `delete_repository_permissions_policy`: Deletes the resource policy that is set on a repository.
• `describe_domain`: Returns a `DomainDescription` object that contains information about the requested domain.
• `describe_package`: Returns a `PackageDescription` object that contains details about a package.
• `describe_package_group`: Returns a `PackageGroup` object that contains details about a package group.
• `describe_package_version`: Returns a `PackageVersionDescription` object that contains details about a package version.
• `describe_repository`: Returns a `RepositoryDescription` object that contains detailed information about the requested repository.
• `dispose_package_versions`: Disposes versions of a package. A package version with the status `Disposed` cannot be restored because they have been permanently removed from storage.
• `disassociate_external_connection`: Removes an existing external connection from a repository.
• `get_associated_package_group`: Returns the most closely associated package group to the specified package.
• `get_authorization_token`: Generates a temporary authorization token for accessing repositories in the domain. The token expires the authorization period has passed. The default authorization period is 12 hours and can be customized to any length with a maximum of 12 hours.
- get_domain_permissions_policy: Returns the policy of a resource that is attached to the specified domain.
- get_package_version_asset: Returns the contents of an asset that is in a package version.
- get_package_version_readme: Gets the readme file or descriptive text for a package version.
- get_repository_endpoint: Returns the endpoint of a repository for a specific package format. A repository has one endpoint for each package format:
  - generic
  - maven
  - npm
  - nuget
  - pypi
  - ruby
  - swift
- get_repository_permissions_policy: Returns the resource policy that is set on a repository.
- list_allowed_repositories_for_group: Lists the allowed repositories for a package group that has origin configuration set to ALLOW_SPECIFIC_REPOSITORIES.
- list_associated_packages: Returns a list of packages associated with the requested package group.
- list_domains: Returns a list of DomainSummary objects. Each returned DomainSummary object contains information about a domain.
- list_packages: Lists the packages in a repository.
- list_package_groups: Returns a list of package groups in the requested domain.
- list_package_version_assets: Lists the assets for a given package version.
- list_package_version_dependencies: Returns a list of the direct dependencies for a package version.
- list_package_versions: Returns a list of package versions for a specified package in a repository.
- list_repositories: Returns a list of repositories owned by the Amazon Web Services account that called this method.
- list_repositories_in_domain: Returns a list of the repositories in a domain.
- list_sub_package_groups: Returns a list of direct children of the specified package group.
- publish_package_version: Creates a new package version containing one or more assets.
- put_domain_permissions_policy: Attaches a resource policy to a domain.
- put_package_origin_configuration: Sets the package origin configuration for a package, which determine how new versions of the package can be added to a specific repository.
- put_repository_permissions_policy: Sets the resource policy on a repository that specifies permissions to access it.
- update_package_group: Updates a package group. This API cannot be used to update a package group’s origin configuration or pattern.
• **update_package_group_origin_configuration**: Updates the package origin configuration for a package group.
• **update_package_versions_status**: Updates the status of one or more versions of a package.
• **update_repository**: Updates the properties of a repository.

Usage

codeartifact(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

credential Optional configuration of credentials, endpoint, and/or region.

  config
    • **credentials**:
      – **creds**:
        • **access_key_id**: AWS access key ID
        • **secret_access_key**: AWS secret access key
        • **session_token**: AWS temporary session token
      – **profile**: The name of a profile to use. If not given, then the default profile is used.
      – **anonymous**: Set anonymous credentials.
    • **endpoint**: The complete URL to use for the constructed client.
    • **region**: The AWS Region used in instantiating the client.
    • **close_connection**: Immediately close all HTTP connections.
    • **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
    • **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
    • **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

credentials Optional credentials shorthand for the config parameter

  • **creds**:
    • **access_key_id**: AWS access key ID
    • **secret_access_key**: AWS secret access key
    • **session_token**: AWS temporary session token
  • **profile**: The name of a profile to use. If not given, then the default profile is used.
  • **anonymous**: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.
Value

A client for the service. You can call the service's operations using syntax like `svc$operation(...)`, where `svc` is the name you've assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- codeartifact(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `associate_external_connection` Adds an existing external connection to a repository
- `copy_package_versions` Copies package versions from one repository to another repository in the same domain
- `create_domain` Creates a domain
- `create_package_group` Creates a package group
- `create_repository` Creates a repository
- `delete_domain` Deletes a domain
- `delete_domain_permissions_policy` Deletes the resource policy set on a domain
- `delete_package` Deletes a package and all associated package versions
delete_package_group Deletes a package group
delete_package_versions Deletes one or more versions of a package
delete_repository Deletes a repository
delete_repository_permissions_policy Deletes the resource policy that is set on a repository
describe_domain Returns a DomainDescription object that contains information about the requested domain
describe_package Returns a PackageDescription object that contains information about the requested package
describe_package_group Returns a PackageGroupDescription object that contains information about the requested package group
describe_package_version Returns a PackageVersionDescription object that contains information about the requested package version
describe_repository Returns a RepositoryDescription object that contains detailed information about the requested repository
disassociate_external_connection Removes an existing external connection from a repository
dispose_package_versions Deletes the assets in package versions and sets the package versions’ status to Disposed
disassociate_package_group Returns the most closely associated package group to the specified package
get_associated_package_group Generates a temporary authorization token for accessing repositories in the domain
get_domain_permissions_policy Returns the resource policy attached to the specified domain
get_package_version_asset Returns an asset (or file) that is in a package
get_package_version_readme Gets the readme file or descriptive text for a package version
get_repository_endpoint Returns the endpoint of a repository for a specific package format
get_repository_permissions_policy Returns the resource policy that is set on a repository
list_allowed_repositories_for_group Lists the repositories in the added repositories list of the specified restriction type for a package group
list_associated_packages Returns a list of packages associated with the requested package group
list_domains Returns a list of DomainSummary objects for all domains owned by the Amazon Web Services account that makes this call
list_package_groups Returns a list of package groups in the requested domain
list_packages Returns a list of PackageSummary objects for packages in a repository that match the request parameters
list_package_version_assets Returns a list of AssetSummary objects for assets in a package version
list_package_version_dependencies Returns the direct dependencies for a package version
list_package_versions Returns a list of PackageVersionSummary objects for package versions in a repository that match the request parameters
list_repositories Returns a list of RepositorySummary objects
list_repositories_in_domain Returns a list of RepositorySummary objects
tag_resource Adds or updates tags for a resource in CodeArtifact
untag_resource Removes tags from a resource in CodeArtifact
update_package_group Updates a package group
update_package_group_origin_configuration Updates the package origin configuration for a package group
update_package_versions_status Updates the status of one or more versions of a package
update_repository Updates the properties of a repository

Examples

## Not run:
svc <- codeartifact()
svc$associate_external_connection(
  Foo = 123
)
Description

CodeBuild

CodeBuild is a fully managed build service in the cloud. CodeBuild compiles your source code, runs unit tests, and produces artifacts that are ready to deploy. CodeBuild eliminates the need to provision, manage, and scale your own build servers. It provides prepackaged build environments for the most popular programming languages and build tools, such as Apache Maven, Gradle, and more. You can also fully customize build environments in CodeBuild to use your own build tools. CodeBuild scales automatically to meet peak build requests. You pay only for the build time you consume. For more information about CodeBuild, see the CodeBuild User Guide.

Usage

codebuild(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)

Arguments

Optional configuration of credentials, endpoint, and/or region.

- **credentials:**
  - **creds:**
    - **access_key_id:** AWS access key ID
    - **secret_access_key:** AWS secret access key
    - **session_token:** AWS temporary session token
  - **profile:** The name of a profile to use. If not given, then the default profile is used.
  - **anonymous:** Set anonymous credentials.
- **endpoint:** The complete URL to use for the constructed client.
- **region:** The AWS Region used in instantiating the client.
- **close_connection:** Immediately close all HTTP connections.
- **timeout:** The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style:** Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy

**credentials** Optional credentials shorthand for the config parameter

• **creds**:
  – **access_key_id**: AWS access key ID
  – **secret_access_key**: AWS secret access key
  – **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- codebuild(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    profile = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
```
```python
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

batch_delete_builds Deletes one or more builds
batch_get_build_batches Retrieves information about one or more batch builds
batch_get_builds Gets information about one or more builds
batch_get_fleets Gets information about one or more compute fleets
batch_get_projects Gets information about one or more build projects
batch_get_report_groups Returns an array of report groups
batch_get_reports For an existing CodeBuild build project that has its source code stored in a GitHub or Bitbucket repository, enables CodeBuild to start rebuilding the source code every time a code change is pushed to the repository
create_fleet Creates a compute fleet
create_project Creates a build project
create_report_group Creates a report group
delete_fleet Deletes a compute fleet
delete_build_batch Deletes a batch build
delete_project Deletes a build project
delete_report Deletes a report
delete_report_group Deletes a report group
delete_resource_policy Deletes a resource policy that is identified by its resource ARN
delete_source_credentials Deletes a set of GitHub, GitHub Enterprise, or Bitbucket source credentials
delete_webhook For an existing CodeBuild build project that has its source code stored in a GitHub or Bitbucket repository, stops CodeBuild from rebuilding the source code every time a code change is pushed to the repository
describe_code_coverages For an existing CodeBuild build project that has its source code stored in a GitHub or Bitbucket repository, for a build that is currently in progress, or for a build that is completed and is being processed, retrieves information about code coverage reports for the build
describe_test_cases Returns a list of details about test cases for a report
describe_report_groups Returns an array of report groups
describe_test_cases
get_report_group_trend Analyzes and accumulates test report values for the specified test reports
get_resource_policy Gets a resource policy that is identified by its resource ARN
import_source_credentials Imports the source repository credentials for an CodeBuild project that has its source code stored in a GitHub, GitHub Enterprise, or Bitbucket repository
invalidate_project_cache Resets the cache for a project
list_build_batches Retrieves the identifiers of your build batches in the current region
list_build_batches_for_project Retrieves the identifiers of the build batches for a specific project
list_builds Gets a list of build IDs, with each build ID representing a single build
list_builds_for_project Gets a list of build identifiers for the specified build project, with each build identifier representing a single build
list_curated_environment_images Gets information about Docker images that are managed by CodeBuild
list_fleets Gets a list of compute fleet names with each compute fleet name representing a single compute fleet
list_fleets_for_project Gets a list ARNs for the report groups in the current Amazon Web Services account
list_proyectos
list_projects Gets a list of project names, with each project name representing a single project
list_report_groups
list_reports
list_reports_for_report_group
list_shared_projects
list_shared_report_groups
list_source_credentials
put_resource_policy Stores a resource policy for the ARN of a Project or ReportGroup object
retry_build
```
retry_build_batch
start_build
start_build_batch
stop_build
stop_build_batch
update_fleet
update_project
update_project_visibility
update_report_group
update_webhook

Restarts a failed batch build
Starts running a build with the settings defined in the project
Starts a batch build for a project
Attempts to stop running a build
Stops a running batch build
Updates a compute fleet
Changes the settings of a build project
Changes the public visibility for a project
Updates a report group
Updates the webhook associated with an CodeBuild build project

Examples

## Not run:
svc <- codebuild()
svc$batch_delete_builds(
  Foo = 123
)

## End(Not run)

codecatalyst  Amazon CodeCatalyst

Description

Welcome to the Amazon CodeCatalyst API reference. This reference provides descriptions of operations and data types for Amazon CodeCatalyst. You can use the Amazon CodeCatalyst API to work with the following objects.

Spaces, by calling the following:

- delete_space, which deletes a space.
- get_space, which returns information about a space.
- get_subscription, which returns information about the Amazon Web Services account used for billing purposes and the billing plan for the space.
- list_spaces, which retrieves a list of spaces.
- update_space, which changes one or more values for a space.

Projects, by calling the following:

- create_project which creates a project in a specified space.
- get_project, which returns information about a project.
- list_projects, which retrieves a list of projects in a space.
Users, by calling the following:

- `get_user_details`, which returns information about a user in Amazon CodeCatalyst.

Source repositories, by calling the following:

- `create_source_repository`, which creates an empty Git-based source repository in a specified project.
- `create_source_repository_branch`, which creates a branch in a specified repository where you can work on code.
- `delete_source_repository`, which deletes a source repository.
- `get_source_repository`, which returns information about a source repository.
- `get_source_repository_clone_urls`, which returns information about the URLs that can be used with a Git client to clone a source repository.
- `list_source_repositories`, which retrieves a list of source repositories in a project.
- `list_source_repository_branches`, which retrieves a list of branches in a source repository.

Dev Environments and the Amazon Web Services Toolkits, by calling the following:

- `create_dev_environment`, which creates a Dev Environment, where you can quickly work on the code stored in the source repositories of your project.
- `delete_dev_environment`, which deletes a Dev Environment.
- `get_dev_environment`, which returns information about a Dev Environment.
- `list_dev_environments`, which retrieves a list of Dev Environments in a project.
- `list_dev_environment_sessions`, which retrieves a list of active Dev Environment sessions in a project.
- `start_dev_environment`, which starts a specified Dev Environment and puts it into an active state.
- `start_dev_environment_session`, which starts a session to a specified Dev Environment.
- `stop_dev_environment`, which stops a specified Dev Environment and puts it into an stopped state.
- `stop_dev_environment_session`, which stops a session for a specified Dev Environment.
- `update_dev_environment`, which changes one or more values for a Dev Environment.

Workflows, by calling the following:

- `get_workflow`, which returns information about a workflow.
- `get_workflow_run`, which returns information about a specified run of a workflow.
- `list_workflow_runs`, which retrieves a list of runs of a specified workflow.
- `list_workflows`, which retrieves a list of workflows in a specified project.
- `start_workflow_run`, which starts a run of a specified workflow.

Security, activity, and resource management in Amazon CodeCatalyst, by calling the following:

- `create_access_token`, which creates a personal access token (PAT) for the current user.
• delete_access_token, which deletes a specified personal access token (PAT).
• list_access_tokens, which lists all personal access tokens (PATs) associated with a user.
• list_event_logs, which retrieves a list of events that occurred during a specified time period in a space.
• verify_session, which verifies whether the calling user has a valid Amazon CodeCatalyst login and session.

If you are using the Amazon CodeCatalyst APIs with an SDK or the CLI, you must configure your computer to work with Amazon CodeCatalyst and single sign-on (SSO). For more information, see Setting up to use the CLI with Amazon CodeCatalyst and the SSO documentation for your SDK.

Usage

codecatalyst()
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

config Optional configuration of credentials, endpoint, and/or region.
    • credentials:
        – creds:
            * access_key_id: AWS access key ID
            * secret_access_key: AWS secret access key
            * session_token: AWS temporary session token
        – profile: The name of a profile to use. If not given, then the default profile is used.
        – anonymous: Set anonymous credentials.
    • endpoint: The complete URL to use for the constructed client.
    • region: The AWS Region used in instantiating the client.
    • close_connection: Immediately close all HTTP connections.
    • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
    • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
    • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ende.html

credentials Optional credentials shorthand for the config parameter
    • creds:
        – access_key_id: AWS access key ID
        – secret_access_key: AWS secret access key
- **session_token**: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.

- **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**
A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
csvc <- codecatalyst(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

**Operations**
create_access_token
create_dev_environment
create_project
create_source_repository
create_source_repository_branch
delete_access_token
delete_dev_environment
delete_project
delete_source_repository
delete_space
get_dev_environment
get_project
get_source_repository
get_source_repository_clone_urls
get_space
get_subscription
get_user_details
get_workflow
get_workflow_run
list_access_tokens
list_dev_environments
list_dev_environment_sessions
list_event_logs
list_projects
list_source_repositories
list_source_repository_branches
list_spaces
list_workflow_runs
list_workflows
start_dev_environment
start_dev_environment_session
start_workflow_run
stop_dev_environment
stop_dev_environment_session
update_dev_environment
update_project
update_space
verify_session

create_access_token
Creates a personal access token (PAT) for the current user
create_dev_environment
Creates a Dev Environment in Amazon CodeCatalyst, a cloud-based development environment that you can use to quickly work on the code stored in the source repositories of your project
create_project
Creates a project in a specified space
create_source_repository
Creates an empty Git-based source repository in a specified project
create_source_repository_branch
Creates a branch in a specified source repository in Amazon CodeCatalyst
delete_access_token
Deletes a specified personal access token (PAT)
delete_dev_environment
Deletes a Dev Environment
delete_project
Deletes a project in a space
delete_source_repository
Deletes a source repository in Amazon CodeCatalyst
delete_space
Deletes a space
get_dev_environment
Returns information about a Dev Environment for a source repository in a project
get_project
Returns information about a project
get_source_repository
Returns information about a source repository
get_source_repository_clone_urls
Returns information about the URLs that can be used with a Git client to clone a source repository
get_space
Returns information about a space
get_subscription
Returns information about the Amazon Web Services account used for billing purposes and the billing plan for the space
get_user_details
Returns information about a user
get_workflow
Returns information about a workflow
list_access_tokens
Lists all personal access tokens (PATs) associated with the user who calls the API
list_dev_environments
Retrieves a list of Dev Environments in a project
list_dev_environment_sessions
Retrieves a list of active sessions for a Dev Environment in a project
list_event_logs
Retrieves a list of events that occurred during a specific time in a space
list_projects
Retrieves a list of projects
list_source_repositories
Retrieves a list of source repositories in a project
list_source_repository_branches
Retrieves a list of branches in a specified source repository
list_spaces
Retrieves a list of spaces
list_workflow_runs
Retrieves a list of workflow runs of a specified workflow
list_workflows
Retrieves a list of workflows in a specified project
start_dev_environment
Starts a specified Dev Environment and puts it into an active state
start_dev_environment_session
Starts a session for a specified Dev Environment
start_workflow_run
 Begins a run of a specified workflow
stop_dev_environment
Pauses a specified Dev Environment and places it in a non-running state
stop_dev_environment_session
 Stops a session for a specified Dev Environment
update_dev_environment
Changes one or more values for a Dev Environment
update_project
Changes one or more values for a project
update_space
Changes one or more values for a space
verify_session
Verifies whether the calling user has a valid Amazon CodeCatalyst login and session

Examples

```r
## Not run:
svc <- codecatalyst()
svc$create_access_token(
    Foo = 123
)
```
## End(Not run)

---

### Description

**CodeCommit**

This is the *CodeCommit API Reference*. This reference provides descriptions of the operations and data types for CodeCommit API along with usage examples.

You can use the CodeCommit API to work with the following objects:

Repositories, by calling the following:

- `batch_get_repositories`, which returns information about one or more repositories associated with your Amazon Web Services account.
- `create_repository`, which creates an CodeCommit repository.
- `delete_repository`, which deletes an CodeCommit repository.
- `get_repository`, which returns information about a specified repository.
- `list_repositories`, which lists all CodeCommit repositories associated with your Amazon Web Services account.
- `update_repository_description`, which sets or updates the description of the repository.
- `update_repository_encryption_key`, which updates the Key Management Service encryption key used to encrypt and decrypt a repository.
- `update_repository_name`, which changes the name of the repository. If you change the name of a repository, no other users of that repository can access it until you send them the new HTTPS or SSH URL to use.

Branches, by calling the following:

- `create_branch`, which creates a branch in a specified repository.
- `delete_branch`, which deletes the specified branch in a repository unless it is the default branch.
- `get_branch`, which returns information about a specified branch.
- `list_branches`, which lists all branches for a specified repository.
- `update_default_branch`, which changes the default branch for a repository.

Files, by calling the following:

- `delete_file`, which deletes the content of a specified file from a specified branch.
- `get_blob`, which returns the base-64 encoded content of an individual Git blob object in a repository.
- `get_file`, which returns the base-64 encoded content of a specified file.
• `get_folder`, which returns the contents of a specified folder or directory.
• `list_file_commit_history`, which retrieves a list of commits and changes to a specified file.
• `put_file`, which adds or modifies a single file in a specified repository and branch.

Commits, by calling the following:

• `batch_get_commits`, which returns information about one or more commits in a repository.
• `create_commit`, which creates a commit for changes to a repository.
• `get_commit`, which returns information about a commit, including commit messages and author and committer information.
• `get_differences`, which returns information about the differences in a valid commit specifier (such as a branch, tag, HEAD, commit ID, or other fully qualified reference).

Merges, by calling the following:

• `batch_describe_merge_conflicts`, which returns information about conflicts in a merge between commits in a repository.
• `create_unreferenced_merge_commit`, which creates an unreferenced commit between two branches or commits for the purpose of comparing them and identifying any potential conflicts.
• `describe_merge_conflicts`, which returns information about merge conflicts between the base, source, and destination versions of a file in a potential merge.
• `get_merge_commit`, which returns information about the merge between a source and destination commit.
• `get_merge_conflicts`, which returns information about merge conflicts between the source and destination branch in a pull request.
• `get_merge_options`, which returns information about the available merge options between two branches or commit specifiers.
• `merge_branches_by_fast_forward`, which merges two branches using the fast-forward merge option.
• `merge_branches_by_squash`, which merges two branches using the squash merge option.
• `merge_branches_by_three_way`, which merges two branches using the three-way merge option.

Pull requests, by calling the following:

• `create_pull_request`, which creates a pull request in a specified repository.
• `create_pull_request_approval_rule`, which creates an approval rule for a specified pull request.
• `delete_pull_request_approval_rule`, which deletes an approval rule for a specified pull request.
• `describe_pull_request_events`, which returns information about one or more pull request events.
• `evaluate_pull_request_approval_rules`, which evaluates whether a pull request has met all the conditions specified in its associated approval rules.
• `get_comments_for_pull_request`, which returns information about comments on a specified pull request.

• `get_pull_request`, which returns information about a specified pull request.

• `get_pull_request_approval_states`, which returns information about the approval states for a specified pull request.

• `get_pull_request_override_state`, which returns information about whether approval rules have been set aside (overridden) for a pull request, and if so, the Amazon Resource Name (ARN) of the user or identity that overrode the rules and their requirements for the pull request.

• `list_pull_requests`, which lists all pull requests for a repository.

• `merge_pull_request_by_fast_forward`, which merges the source destination branch of a pull request into the specified destination branch for that pull request using the fast-forward merge option.

• `merge_pull_request_by_squash`, which merges the source destination branch of a pull request into the specified destination branch for that pull request using the squash merge option.

• `merge_pull_request_by_three_way`, which merges the source destination branch of a pull request into the specified destination branch for that pull request using the three-way merge option.

• `override_pull_request_approval_rules`, which sets aside all approval rule requirements for a pull request.

• `post_comment_for_pull_request`, which posts a comment to a pull request at the specified line, file, or request.

• `update_pull_request_approval_rule_content`, which updates the structure of an approval rule for a pull request.

• `update_pull_request_approval_state`, which updates the state of an approval on a pull request.

• `update_pull_request_description`, which updates the description of a pull request.

• `update_pull_request_status`, which updates the status of a pull request.

• `update_pull_request_title`, which updates the title of a pull request.

Approval rule templates, by calling the following:

• `associate_approval_rule_template_with_repository`, which associates a template with a specified repository. After the template is associated with a repository, CodeCommit creates approval rules that match the template conditions on every pull request created in the specified repository.

• `batch_associate_approval_rule_template_with_repositories`, which associates a template with one or more specified repositories. After the template is associated with a repository, CodeCommit creates approval rules that match the template conditions on every pull request created in the specified repositories.

• `batch_disassociate_approval_rule_template_from_repositories`, which removes the association between a template and specified repositories so that approval rules based on the template are not automatically created when pull requests are created in those repositories.
• create_approval_rule_template, which creates a template for approval rules that can then be associated with one or more repositories in your Amazon Web Services account.

• delete_approval_rule_template, which deletes the specified template. It does not remove approval rules on pull requests already created with the template.

• disassociate_approval_rule_template_from_repository, which removes the association between a template and a repository so that approval rules based on the template are not automatically created when pull requests are created in the specified repository.

• get_approval_rule_template, which returns information about an approval rule template.

• list_approval_rule_templates, which lists all approval rule templates in the Amazon Web Services Region in your Amazon Web Services account.

• list_associated_approval_rule_templates_for_repository, which lists all approval rule templates that are associated with a specified repository.

• list_repositories_for_approval_rule_template, which lists all repositories associated with the specified approval rule template.

• update_approval_rule_template_description, which updates the description of an approval rule template.

• update_approval_rule_template_name, which updates the name of an approval rule template.

• update_approval_rule_template_content, which updates the content of an approval rule template.

Comments in a repository, by calling the following:

• delete_comment_content, which deletes the content of a comment on a commit in a repository.

• get_comment, which returns information about a comment on a commit.

• get_comment_reactions, which returns information about emoji reactions to comments.

• get_comments_for_compared_commit, which returns information about comments on the comparison between two commit specifiers in a repository.

• post_comment_for_compared_commit, which creates a comment on the comparison between two commit specifiers in a repository.

• post_comment_reply, which creates a reply to a comment.

• put_comment_reaction, which creates or updates an emoji reaction to a comment.

• update_comment, which updates the content of a comment on a commit in a repository.

Tags used to tag resources in CodeCommit (not Git tags), by calling the following:

• list_tags_for_resource, which gets information about Amazon Web Service tags for a specified Amazon Resource Name (ARN) in CodeCommit.

• tag_resource, which adds or updates tags for a resource in CodeCommit.

• untag_resource, which removes tags for a resource in CodeCommit.

Triggers, by calling the following:

• get_repository_triggers, which returns information about triggers configured for a repository.
• put_repository_triggers, which replaces all triggers for a repository and can be used to create or delete triggers.
• test_repository_triggers, which tests the functionality of a repository trigger by sending data to the trigger target.

For information about how to use CodeCommit, see the CodeCommit User Guide.

Usage

```r
codecommit(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**: 
    - access_key_id: AWS access key ID
    - secret_access_key: AWS secret access key
    - session_token: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.
- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**: 
    - access_key_id: AWS access key ID
    - secret_access_key: AWS secret access key
    - session_token: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

- **endpoint**: Optional shorthand for complete URL to use for the constructed client.
- **region**: Optional shorthand for AWS Region used in instantiating the client.
Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- codecommit(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `associate_approval_rule_template_with_repository` Creates an association between an approval rule template and a repository.
- `batch_associate_approval_rule_template_with_repositories` Creates an association between an approval rule template and multiple repositories.
- `batch_describe_merge_conflicts` Returns information about one or more merge conflicts in the attempted merge of two commit specifiers using the squash or three-way merge strategy.
- `batch_disassociate_approval_rule_template_from_repositories` Removes the association between an approval rule template and one or more repositories.
- `batch_get_commits` Returns information about the contents of one or more commits in a repository.
- `batch_get_repositories` Returns information about one or more repositories.
- `create_approval_rule_template` Creates a template for approval rules that can then be associated with repositories.
- `create.branch` Creates a branch in a repository and points the branch to a commit.
create_commit
create_pull_request
create_pull_requestApprovalRule
create_repository
create_unreferenced_merge_commit
delete_approval_rule_template
delete_branch
delete_comment_content
delete_file
delete_pull_requestApprovalRule
delete_repository
describe_merge_conflicts
describe_pull_request_events
disassociate_approval_rule_template_from_repository
evaluate_pull_requestApprovalRules
get_approval_rule_template
get_blob
get_branch
get_comment
get_comment_reactions
get_comments_for_compared_commit
get_comments_for_pull_request
get_commit
get_differences
get_file
get_folder
get_merge_commit
get_merge_conflicts
get_merge_options
get_pull_request
get_pull_requestApprovalStates
get_pull_requestOverrideState
get_repository
get_repository_triggers
list_approval_rule_templates
list_associated_approval_rule_templates_for_repository
list_branches
list_file_commit_history
list_pull_requests
list_repositories
list_repositories_for_approval_rule_template
list_tags_for_resource
merge_branches_by_fast_forward
merge_branches_by_squash
merge_branches_by_three_way
merge_pull_request_by_fast_forward
merge_pull_request_by_squash
merge_pull_request_by_three_way

Creates a commit for a repository on the tip of a specified branch.
Creates a pull request in the specified repository.
Creates an approval rule for a pull request.
Creates a new, empty repository.
Creates an unreferenced commit that represents the result of an
unspecified merge.
deletes a specified approval rule template.
deletes a branch from a repository, unless that branch is the default branch.
deletes the content of a comment made on a change, file, or repository.
deletes a specified file from a specified branch.
deletes an approval rule from a specified pull request.
deletes a repository.
returns information about one or more merge conflicts in the
repository.
returns information about one or more pull request events.
removes the association between a template and a repository.
evaluates whether a pull request has met all the conditions specified
in its associated approval rules.
returns information about a specified approval rule template.
returns the base-64 encoded content of an individual blob in a
repository.
returns information about a repository branch, including its ID.
returns the content of a comment made on a change, file, or repository.
returns information about reactions to a specified comment ID.
returns information about comments made on the comparison
between two commits.
returns comments made on a pull request.
returns information about a commit, including commit message.
returns information about the differences in a valid commit specifier.
returns the base-64 encoded contents of a specified file and its
metadata.
returns information about a specified merge commit.
returns information about merge conflicts between the before and
after commit.
returns information about the merge options available for merging.
gets information about a pull request in a specified repository.
gets information about the approval states for a specified pull request.
returns information about whether approval rules have been
evaluated for a pull request.
returns information about a repository.
gets information about triggers configured for a repository.
lists all approval rule templates in the specified Amazon Web
Services Region.
lists all approval rule templates that are associated with a specific
repository.
gets information about one or more branches in a repository.
retrieves a list of commits and changes to a specified file.
returns a list of pull requests for a specified repository.
gets information about one or more repositories.
lists all repositories associated with the specified approval rule.
gets information about Amazon Web ServiceStagS.
merges two branches using the fast-forward merge strategy.
merges two branches using the squash merge strategy.
merges two specified branches using the three-way merge strategy.
Attempts to merge the source commit of a pull request into the
destination branch.
Attempts to merge the source commit of a pull request into the
destination branch using the specified merge strategy.
override_pull_request_approval_rules
post_comment_for_compared_commit
post_comment_for_pull_request
post_comment_reply
put_comment_reaction
put_file
put_repository_triggers
tag_resource
test_repository_triggers
untag_resource
update_approval_rule_template_content
update_approval_rule_template_description
update_approval_rule_template_name
update_comment
update_default_branch
update_pull_request_approval_rule_content
update_pull_request_approval_state
update_pull_request_description
update_pull_request_status
update_pull_request_title
update_repository_description
update_repository_encryption_key
update_repository_name

Examples

## Not run:
svc <- codecommit()
svc$associate_approval_rule_template_with_repository(
  Foo = 123
)

## End(Not run)

codedeploy

AWS CodeDeploy

Description

CodeDeploy is a deployment service that automates application deployments to Amazon EC2 instances, on-premises instances running in your own facility, serverless Lambda functions, or applications in an Amazon ECS service.

You can deploy a nearly unlimited variety of application content, such as an updated Lambda function, updated applications in an Amazon ECS service, code, web and configuration files, executables, packages, scripts, multimedia files, and so on. CodeDeploy can deploy application content that
stored in Amazon S3 buckets, GitHub repositories, or Bitbucket repositories. You do not need to make changes to your existing code before you can use CodeDeploy.

CodeDeploy makes it easier for you to rapidly release new features, helps you avoid downtime during application deployment, and handles the complexity of updating your applications, without many of the risks associated with error-prone manual deployments.

**CodeDeploy Components**

Use the information in this guide to help you work with the following CodeDeploy components:

- **Application**: A name that uniquely identifies the application you want to deploy. CodeDeploy uses this name, which functions as a container, to ensure the correct combination of revision, deployment configuration, and deployment group are referenced during a deployment.

- **Deployment group**: A set of individual instances, CodeDeploy Lambda deployment configuration settings, or an Amazon ECS service and network details. A Lambda deployment group specifies how to route traffic to a new version of a Lambda function. An Amazon ECS deployment group specifies the service created in Amazon ECS to deploy, a load balancer, and a listener to reroute production traffic to an updated containerized application. An Amazon EC2/On-premises deployment group contains individually tagged instances, Amazon EC2 instances in Amazon EC2 Auto Scaling groups, or both. All deployment groups can specify optional trigger, alarm, and rollback settings.

- **Deployment configuration**: A set of deployment rules and deployment success and failure conditions used by CodeDeploy during a deployment.

- **Deployment**: The process and the components used when updating a Lambda function, a containerized application in an Amazon ECS service, or of installing content on one or more instances.

- **Application revisions**: For an Lambda deployment, this is an AppSpec file that specifies the Lambda function to be updated and one or more functions to validate deployment lifecycle events. For an Amazon ECS deployment, this is an AppSpec file that specifies the Amazon ECS task definition, container, and port where production traffic is rerouted. For an EC2/On-premises deployment, this is an archive file that contains source content—source code, webpages, executable files, and deployment scripts—along with an AppSpec file. Revisions are stored in Amazon S3 buckets or GitHub repositories. For Amazon S3, a revision is uniquely identified by its Amazon S3 object key and its ETag, version, or both. For GitHub, a revision is uniquely identified by its commit ID.

This guide also contains information to help you get details about the instances in your deployments, to make on-premises instances available for CodeDeploy deployments, to get details about a Lambda function deployment, and to get details about Amazon ECS service deployments.

**CodeDeploy Information Resources**

- CodeDeploy User Guide
- CodeDeploy API Reference Guide
- CLI Reference for CodeDeploy
- CodeDeploy Developer Forum
Usage

codedeploy(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

config Optional configuration of credentials, endpoint, and/or region.

  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.

  • endpoint: The complete URL to use for the constructed client.

  • region: The AWS Region used in instantiating the client.

  • close_connection: Immediately close all HTTP connections.

  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html

credentials Optional credentials shorthand for the config parameter

  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token

  • profile: The name of a profile to use. If not given, then the default profile is used.

  • anonymous: Set anonymous credentials.

close_connection Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

svc <- codedeploy(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
))

Operations

add_tags_to_on_premises_instances  Adds tags to on-premises instances
batch_get_application_revisions    Gets information about one or more application revisions
batch_get_applications             Gets information about one or more applications
batch_get_deployment_groups        Gets information about one or more deployment groups
batch_get_deployment_instances     This method works, but is deprecated
batch_get_deployments              Gets information about one or more deployments
batch_get_deployment_targets       Returns an array of one or more targets associated with a deployment
batch_get_on_premises_instances    Gets information about one or more on-premises instances
continue_deployment               For a blue/green deployment, starts the process of rerouting traffic from instances in the original environment to instances in the replacement environment without waiting for a specified wait time to elapse
create_application                Creates an application
create_deployment                 Deploys an application revision through the specified deployment group
create_deployment_config          Creates a deployment configuration
create_deployment_group           Creates a deployment group to which application revisions are deployed
delete_application               Deletes an application
<table>
<thead>
<tr>
<th>Function Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>delete_deployment_config</td>
<td>Deletes a deployment configuration</td>
</tr>
<tr>
<td>delete_deployment_group</td>
<td>Deletes a deployment group</td>
</tr>
<tr>
<td>delete_git_hub_account_token</td>
<td>Deletes a GitHub account connection</td>
</tr>
<tr>
<td>delete_resources_by_external_id</td>
<td>Deletes resources linked to an external ID</td>
</tr>
<tr>
<td>deregister_on_premises_instance</td>
<td>Deregisters an on-premises instance</td>
</tr>
<tr>
<td>get_application</td>
<td>Gets information about an application</td>
</tr>
<tr>
<td>get_application_revision</td>
<td>Gets information about an application revision</td>
</tr>
<tr>
<td>get_deployment</td>
<td>Gets information about a deployment</td>
</tr>
<tr>
<td>get_deployment_config</td>
<td>Gets information about a deployment configuration</td>
</tr>
<tr>
<td>get_deployment_group</td>
<td>Gets information about a deployment group</td>
</tr>
<tr>
<td>get_deployment_instance</td>
<td>Gets information about an instance as part of a deployment</td>
</tr>
<tr>
<td>get_deployment_target</td>
<td>Returns information about a deployment target</td>
</tr>
<tr>
<td>get_on_premises_instance</td>
<td>Gets information about an on-premises instance</td>
</tr>
<tr>
<td>list_application_revisions</td>
<td>Lists information about revisions for an application</td>
</tr>
<tr>
<td>list_applications</td>
<td>Lists the applications registered with the user or Amazon Web Services account</td>
</tr>
<tr>
<td>list_deployment_configs</td>
<td>Lists the deployment configurations with the user or Amazon Web Services account</td>
</tr>
<tr>
<td>list_deployment_groups</td>
<td>Lists the deployment groups for an application registered with the Amazon Web Services account</td>
</tr>
<tr>
<td>list_deployment_instances</td>
<td>Lists the deployments in a deployment group for an application registered with the Amazon Web Services account</td>
</tr>
<tr>
<td>list_deployments</td>
<td>Returns an array of target IDs that are associated a deployment</td>
</tr>
<tr>
<td>list_deployment_targets</td>
<td>Lists the names of stored connections to GitHub accounts</td>
</tr>
<tr>
<td>list_on_premises_instances</td>
<td>Gets a list of names for one or more on-premises instances</td>
</tr>
<tr>
<td>list_tags_for_resource</td>
<td>Returns a list of tags for the resource identified by a specified Amazon Resource Name</td>
</tr>
<tr>
<td>put_lifecycle_event_hook_execution_status</td>
<td>Registers with CodeDeploy a revision for the specified application</td>
</tr>
<tr>
<td>register_application_revision</td>
<td>Registers an on-premises instance</td>
</tr>
<tr>
<td>register_on_premises_instance</td>
<td>Removes one or more tags from one or more on-premises instances</td>
</tr>
<tr>
<td>remove_tags_from_on_premises_instances</td>
<td>In a blue/green deployment, overrides any specified wait time and starts terminating instances immediately after the traffic routing is complete</td>
</tr>
<tr>
<td>skip_wait_time_for_instance_termination</td>
<td>Attempts to stop an ongoing deployment</td>
</tr>
<tr>
<td>stop_deployment</td>
<td>Associates the list of tags in the input Tags parameter with the resource identified by a specified Amazon Resource Name</td>
</tr>
<tr>
<td>tag_resource</td>
<td>Disassociates a resource from a list of tags</td>
</tr>
<tr>
<td>untag_resource</td>
<td>Changes the name of an application</td>
</tr>
<tr>
<td>update_application</td>
<td>Changes information about a deployment group</td>
</tr>
</tbody>
</table>

### Examples

```r
## Not run:
svc <- codedeploy()
svc$add_tags_to_on_premises_instances(
  Foo = 123
)

## End(Not run)
```
Description

This section provides documentation for the Amazon CodeGuru Profiler API operations. Amazon CodeGuru Profiler collects runtime performance data from your live applications, and provides recommendations that can help you fine-tune your application performance. Using machine learning algorithms, CodeGuru Profiler can help you find your most expensive lines of code and suggest ways you can improve efficiency and remove CPU bottlenecks.

Amazon CodeGuru Profiler provides different visualizations of profiling data to help you identify what code is running on the CPU, see how much time is consumed, and suggest ways to reduce CPU utilization.

Amazon CodeGuru Profiler currently supports applications written in all Java virtual machine (JVM) languages and Python. While CodeGuru Profiler supports both visualizations and recommendations for applications written in Java, it can also generate visualizations and a subset of recommendations for applications written in other JVM languages and Python.

For more information, see What is Amazon CodeGuru Profiler in the Amazon CodeGuru Profiler User Guide.

Usage

codeguruprofiler(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

config Optional configuration of credentials, endpoint, and/or region.

  credentials:
    • creds:
    - access_key_id: AWS access key ID
    - secret_access_key: AWS secret access key
    - session_token: AWS temporary session token
    - profile: The name of a profile to use. If not given, then the default profile is used.
    - anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

• **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends\(html\]

**credentials** Optional credentials shorthand for the config parameter

• **creds**:
  – **access_key_id**: AWS access key ID
  – **secret_access_key**: AWS secret access key
  – **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
doc <- codeguruprofiler(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string",
  close_connection = "logical",
  timeout = "numeric",
  s3_force_path_style = "logical",
  sts_regional_endpoint = "string"
)
```
secret_access_key = "string",
    session_token = "string"
  ),
  profile = "string",
  anonymous = "logical"
),
  endpoint = "string",
  region = "string"
)

Operations

add_notification_channels Add up to 2 anomaly notifications channels for a profiling group
batch_get_frame_metric_data Returns the time series of values for a requested list of frame metrics from a time period
configure_agent Used by profiler agents to report their current state and to receive remote configuration updates
create_profiling_group Creates a profiling group
delete_profiling_group Deletes a profiling group
describe_profiling_group Returns a ProfilingGroupDescription object that contains information about the requested profiling group
get_findings_report_account_summary Returns a list of FindingsReportSummary objects that contain analysis results for all
get_notification_configuration Get the current configuration for anomaly notifications for a profiling group
get_policy Returns the JSON-formatted resource-based policy on a profiling group
get_profile Gets the aggregated profile of a profiling group for a specified time range
get_recommendations Returns a list of Recommendation objects that contain recommendations for a profiling group for a given time period
list_findings_reports List the available reports for a given profiling group and time range
list_profile_times Lists the start times of the available aggregated profiles of a profiling group for an aggregation period within the specified time range
list_profiling_groups Returns a list of profiling groups
list_tags_for_resource Returns a list of the tags that are assigned to a specified resource
post_agent_profile Submits profiling data to an aggregated profile of a profiling group
put_permission Adds permissions to a profiling group’s resource-based policy that are provided using an action group
remove_notification_channel Remove one anomaly notifications channel for a profiling group
remove_permission Removes permissions from a profiling group’s resource-based policy that are provided using an action group
submit_feedback Sends feedback to CodeGuru Profiler about whether the anomaly detected by the analysis is useful or not
tag_resource Use to assign one or more tags to a resource
untag_resource Use to remove one or more tags from a resource
update_profiling_group Updates a profiling group

Examples

## Not run:
svc <- codeguruprofiler()
svc$add_notification_channels(
  Foo = 123
)

## End(Not run)
Description

This section provides documentation for the Amazon CodeGuru Reviewer API operations. CodeGuru Reviewer is a service that uses program analysis and machine learning to detect potential defects that are difficult for developers to find and recommends fixes in your Java and Python code. By proactively detecting and providing recommendations for addressing code defects and implementing best practices, CodeGuru Reviewer improves the overall quality and maintainability of your code base during the code review stage. For more information about CodeGuru Reviewer, see the Amazon CodeGuru Reviewer User Guide.

To improve the security of your CodeGuru Reviewer API calls, you can establish a private connection between your VPC and CodeGuru Reviewer by creating an interface VPC endpoint. For more information, see CodeGuru Reviewer and interface VPC endpoints (Amazon Web Services PrivateLink) in the Amazon CodeGuru Reviewer User Guide.

Usage

codegurureviewer(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

cfg Optional configuration of credentials, endpoint, and/or region.

  • credentials:
      – creds:
          * access_key_id: AWS access key ID
          * secret_access_key: AWS secret access key
          * session_token: AWS temporary session token
      – profile: The name of a profile to use. If not given, then the default profile is used.
      – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy

**Credentials**

Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.

- **anonymous**: Set anonymous credentials.

**Endpoint**

Optional shorthand for complete URL to use for the constructed client.

**Region**

Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service's operations using syntax like `svc$operation(...)`, where `svc` is the name you've assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- codegurureviewer(
  config = list(  
    credentials = list(  
      creds = list(  
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(  
    creds = list(  
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
```
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

**Operations**

- **associate_repository**
  - Use to associate an Amazon Web Services CodeCommit repository or a repository managed by Amazon Web Services CodeStar Connections with Amazon CodeGuru Reviewer

- **create_code_review**
  - Use to create a code review with a CodeReviewType of RepositoryAnalysis

- **describe_code_review**
  - Returns the metadata associated with the code review along with its status

- **describe_recommendation_feedback**
  - Describes the customer feedback for a CodeGuru Reviewer recommendation

- **describe_repository_association**
  - Returns a RepositoryAssociation object that contains information about the requested repository association

- **disassociate_repository**
  - Removes the association between Amazon CodeGuru Reviewer and a repository

- **list_code_reviews**
  - Lists all the code reviews that the customer has created in the past 90 days

- **list_recommendation_feedback**
  - Returns a list of RecommendationFeedbackSummary objects that contain customer recommendation feedback for all CodeGuru Reviewer users

- **list_recommendations**
  - Returns the list of all recommendations for a completed code review

- **list_repository_associations**
  - Returns a list of RepositoryAssociationSummary objects that contain summary information about repository associations

- **list_tags_for_resource**
  - Returns the list of tags associated with an associated repository resource

- **put_recommendation_feedback**
  - Stores customer feedback for a CodeGuru Reviewer recommendation

- **tag_resource**
  - Adds one or more tags to an associated repository

- **untag_resource**
  - Removes a tag from an associated repository

**Examples**

```r
## Not run:
svc <- codegurureviewer()
svc$associate_repository(
  Foo = 123
)

## End(Not run)
```

---

codegurusecurity

**Amazon CodeGuru Security**

---

**Description**

Amazon CodeGuru Security is in preview release and is subject to change.

This section provides documentation for the Amazon CodeGuru Security API operations. CodeGuru Security is a service that uses program analysis and machine learning to detect security policy violations and vulnerabilities, and recommends ways to address these security risks.

By proactively detecting and providing recommendations for addressing security risks, CodeGuru Security improves the overall security of your application code. For more information about CodeGuru Security, see the [Amazon CodeGuru Security User Guide](#).
Usage

codegurusecurity(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

cfgurusecurity(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**: 
    - `creds`:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
    - `anonymous`: Set anonymous credentials.

- **endpoint**: The complete URL to use for the constructed client.

- **region**: The AWS Region used in instantiating the client.

- **close_connection**: Immediately close all HTTP connections.

- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

- **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.

- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - `profile`: The name of a profile to use. If not given, then the default profile is used.
  - `anonymous`: Set anonymous credentials.

- **endpoint**: Optional shorthand for complete URL to use for the constructed client.

- **region**: Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

svc <- codegurusecurity(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)

Operations

- batch_get_findings: Returns a list of all requested findings
- create_scan: Use to create a scan using code uploaded to an S3 bucket
- create_upload_url: Generates a pre-signed URL and request headers used to upload a code resource
- get_account_configuration: Use to get account level configuration
- get_findings: Returns a list of all findings generated by a particular scan
- get_metrics_summary: Returns top level metrics about an account from a specified date, including number of open findings, the categories with most findings, the scans with most open findings, and scans with most open critical findings
- get_scan: Returns details about a scan, including whether or not a scan has completed
- list_findings_metrics: Returns metrics about all findings in an account within a specified time range
- list_scans: Returns a list of all the standard scans in an account
- list_tags_for_resource: Returns a list of all tags associated with a scan
- tag_resource: Use to add one or more tags to an existing scan
- untag_resource: Use to remove one or more tags from an existing scan
- update_account_configuration: Use to update account-level configuration with an encryption key
Examples

```r
## Not run:
svc <- codegurusecurity()
svc$batch_get_findings(
  Foo = 123
)

## End(Not run)
```

---

**codepipeline**

**AWS CodePipeline**

---

**Description**

CodePipeline

**Overview**

This is the CodePipeline API Reference. This guide provides descriptions of the actions and data types for CodePipeline. Some functionality for your pipeline can only be configured through the API. For more information, see the CodePipeline User Guide.

You can use the CodePipeline API to work with pipelines, stages, actions, and transitions. *Pipelines* are models of automated release processes. Each pipeline is uniquely named, and consists of stages, actions, and transitions.

You can work with pipelines by calling:

- `create_pipeline`, which creates a uniquely named pipeline.
- `delete_pipeline`, which deletes the specified pipeline.
- `get_pipeline`, which returns information about the pipeline structure and pipeline metadata, including the pipeline Amazon Resource Name (ARN).
- `get_pipeline_execution`, which returns information about a specific execution of a pipeline.
- `get_pipeline_state`, which returns information about the current state of the stages and actions of a pipeline.
- `list_action_executions`, which returns action-level details for past executions. The details include full stage and action-level details, including individual action duration, status, any errors that occurred during the execution, and input and output artifact location details.
- `list_pipelines`, which gets a summary of all of the pipelines associated with your account.
- `list_pipeline_executions`, which gets a summary of the most recent executions for a pipeline.
- `start_pipeline_execution`, which runs the most recent revision of an artifact through the pipeline.
• **stop_pipeline_execution**, which stops the specified pipeline execution from continuing through the pipeline.

• **update_pipeline**, which updates a pipeline with edits or changes to the structure of the pipeline.

Pipelines include **stages**. Each stage contains one or more actions that must complete before the next stage begins. A stage results in success or failure. If a stage fails, the pipeline stops at that stage and remains stopped until either a new version of an artifact appears in the source location, or a user takes action to rerun the most recent artifact through the pipeline. You can call **get_pipeline_state**, which displays the status of a pipeline, including the status of stages in the pipeline, or **get_pipeline**, which returns the entire structure of the pipeline, including the stages of that pipeline. For more information about the structure of stages and actions, see CodePipeline Pipeline Structure Reference.

Pipeline stages include **actions** that are categorized into categories such as source or build actions performed in a stage of a pipeline. For example, you can use a source action to import artifacts into a pipeline from a source such as Amazon S3. Like stages, you do not work with actions directly in most cases, but you do define and interact with actions when working with pipeline operations such as **create_pipeline** and **get_pipeline_state**. Valid action categories are:

• Source
• Build
• Test
• Deploy
• Approval
• Invoke

Pipelines also include **transitions**, which allow the transition of artifacts from one stage to the next in a pipeline after the actions in one stage complete.

You can work with transitions by calling:

• **disable_stage_transition**, which prevents artifacts from transitioning to the next stage in a pipeline.
• **enable_stage_transition**, which enables transition of artifacts between stages in a pipeline.

**Using the API to integrate with CodePipeline**

For third-party integrators or developers who want to create their own integrations with CodePipeline, the expected sequence varies from the standard API user. To integrate with CodePipeline, developers need to work with the following items:

**Jobs**, which are instances of an action. For example, a job for a source action might import a revision of an artifact from a source.

You can work with jobs by calling:

• **acknowledge_job**, which confirms whether a job worker has received the specified job.
• **get_job_details**, which returns the details of a job.
• **poll_for_jobs**, which determines whether there are any jobs to act on.
• **put_job_failure_result**, which provides details of a job failure.
• put_job_success_result, which provides details of a job success.

Third party jobs, which are instances of an action created by a partner action and integrated into CodePipeline. Partner actions are created by members of the Amazon Web Services Partner Network.

You can work with third party jobs by calling:

• acknowledge_third_party_job, which confirms whether a job worker has received the specified job.
• get_third_party_job_details, which requests the details of a job for a partner action.
• poll_for_third_party_jobs, which determines whether there are any jobs to act on.
• put_third_party_job_failure_result, which provides details of a job failure.
• put_third_party_job_success_result, which provides details of a job success.

Usage

codepipeline(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

codepipeline

Optional configuration of credentials, endpoint, and/or region.

• **credentials:**
  
  • **creds:**
    
    * access_key_id: AWS access key ID
    * secret_access_key: AWS secret access key
    * session_token: AWS temporary session token
  
  • **profile:** The name of a profile to use. If not given, then the default profile is used.
  
  • **anonymous:** Set anonymous credentials.

• **endpoint:** The complete URL to use for the constructed client.

• **region:** The AWS Region used in instantiating the client.

• **close_connection:** Immediately close all HTTP connections.

• **timeout:** The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

• **s3_force_path_style:** Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

• **sts_regional_endpoint:** Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-entities.html

**credentials** Optional credentials shorthand for the config parameter
• **creds:**
  – **access_key_id:** AWS access key ID
  – **secret_access_key:** AWS secret access key
  – **session_token:** AWS temporary session token
• **profile:** The name of a profile to use. If not given, then the default profile is used.
• **anonymous:** Set anonymous credentials.

  **endpoint**  Optional shorthand for complete URL to use for the constructed client.
  **region**  Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- codepipeline(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```
Operations

acknowledge_job
acknowledge_third_party_job
create_custom_action_type
create_pipeline
delete_custom_action_type
delete_pipeline
delete_webhook
deregister_webhook_with_third_party
disable_stage_transition
enable_stage_transition
get_action_type
get_job_details
get_pipeline
get_pipeline_execution
get_pipeline_state
get_third_party_job_details
list_action_executions
list_action_types
list_pipeline_executions
list_pipelines
list_tags_for_resource
list_webhooks
poll_for_jobs
poll_for_third_party_jobs
put_action_revision
put_approval_result
put_job_failure_result
put_job_success_result
put_third_party_job_failure_result
put_third_party_job_success_result
put_webhook
register_webhook_with_third_party
retry_stage_execution
rollback_stage
start_pipeline_execution
stop_pipeline_execution
tag_resource
untag_resource
update_action_type
update_pipeline

Returns information about a specified job and whether that job has been received by the pipeline worker.
Confirms a job worker has received the specified job.
Creates a new custom action that can be used in all pipelines associated with the Amazon Web Services account.
Creates a pipeline.
Marks a custom action as deleted.
Deletes the specified pipeline.
Deletes a previously created webhook by name.
Removes the connection between the webhook that was created by CodePipeline and the external tool.
Prevents artifacts in a pipeline from transitioning to the next stage in the pipeline.
Enables artifacts in a pipeline to transition to a stage in the pipeline.
Returns information about an action type created for an external provider, where the action type is to be used by customers of the external provider.
Returns information about a job.
Returns the metadata, structure, stages, and actions of a pipeline.
Returns information about an execution of a pipeline, including details about artifacts.
Returns information about the state of a pipeline, including the stages and actions.
Requests the details of a job for a third-party action.
Lists the action executions that have occurred in a pipeline.
Gets a summary of all CodePipeline action types associated with your account.
Gets a summary of the most recent executions for a pipeline.
Gets a summary of all of the pipelines associated with your account.
Gets the set of key-value pairs (metadata) that are used to manage the resource.
Gets a listing of all the webhooks in this Amazon Web Services Region for this account.
Returns information about any jobs for CodePipeline to act on.
Determines whether there are any third party jobs for a job worker to act on.
Provides information to CodePipeline about new revisions to a source.
Provides the response to a manual approval request to CodePipeline.
Represents the failure of a job as returned to the pipeline by a job worker.
Represents the success of a job as returned to the pipeline by a job worker.
Represents the failure of a third party job as returned to the pipeline by a job worker.
Represents the success of a third party job as returned to the pipeline by a job worker.
Defines a webhook and returns a unique webhook URL generated by CodePipeline.
Configures a connection between the webhook that was created and the external tool.
You can retry a stage that has failed without having to run a pipeline again from the beginning.
Rolls back a stage execution.
Starts the specified pipeline.
Stops the specified pipeline execution.
Adds to or modifies the tags of the given resource.
Removes tags from an Amazon Web Services resource.
Updates an action type that was created with any supported integration model, where the action type is to be used by customers of the action type provider.
Updates a specified pipeline with edits or changes to its structure.

Examples

## Not run:
```r
svc <- codepipeline()
svc$acknowledge_job(
  Foo = 123
)

## End(Not run)
```

---

**AWS CodeStar**

**Description**

This is the API reference for AWS CodeStar. This reference provides descriptions of the operations and data types for the AWS CodeStar API along with usage examples.

You can use the AWS CodeStar API to work with:

Projects and their resources, by calling the following:

- `delete_project`, which deletes a project.
- `describe_project`, which lists the attributes of a project.
- `list_projects`, which lists all projects associated with your AWS account.
- `list_resources`, which lists the resources associated with a project.
- `list_tags_for_project`, which lists the tags associated with a project.
- `tag_project`, which adds tags to a project.
- `untag_project`, which removes tags from a project.
- `update_project`, which updates the attributes of a project.

Teams and team members, by calling the following:

- `associate_team_member`, which adds an IAM user to the team for a project.
- `disassociate_team_member`, which removes an IAM user from the team for a project.
- `list_team_members`, which lists all the IAM users in the team for a project, including their roles and attributes.
- `update_team_member`, which updates a team member's attributes in a project.

Users, by calling the following:

- `create_user_profile`, which creates a user profile that contains data associated with the user across all projects.
- `delete_user_profile`, which deletes all user profile information across all projects.
- `describe_user_profile`, which describes the profile of a user.
- `list_user_profiles`, which lists all user profiles.
- `update_user_profile`, which updates the profile for a user.
Usage

codestar(config = list(), credentials = list(), endpoint = NULL, region = NULL)

Arguments

config  Optional configuration of credentials, endpoint, and/or region.

  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials  Optional credentials shorthand for the config parameter

  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

endpoint  Optional shorthand for complete URL to use for the constructed client.

region  Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

svc <- codestar(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)

Operations

associate_team_member Adds an IAM user to the team for an AWS CodeStar project
create_project Creates a project, including project resources
create_user_profile Creates a profile for a user that includes user preferences, such as the display name and email address
delete_project Deletes a project, including project resources
delete_user_profile Deletes a user profile in AWS CodeStar, including all personal preference data associated with that profile
describe_project Describes a project and its resources
describe_user_profile Describes a user in AWS CodeStar and the user attributes across all projects
disassociate_team_member Removes a user from a project
list_projects Lists all projects in AWS CodeStar associated with your AWS account
list_resources Lists resources associated with a project in AWS CodeStar
list_tags_for_project Gets the tags for a project
list_team_members Lists all team members associated with a project
list_user_profiles Lists all the user profiles configured for your AWS account in AWS CodeStar
tag_project Adds tags to a project
### Examples

```r
## Not run:
svc <- codestar()
svc$associate_team_member(
  Foo = 123
)

## End(Not run)
```

### Description

AWS CodeStar Connections

This Amazon Web Services CodeStar Connections API Reference provides descriptions and usage examples of the operations and data types for the Amazon Web Services CodeStar Connections API. You can use the connections API to work with connections and installations. **Connections** are configurations that you use to connect Amazon Web Services resources to external code repositories. Each connection is a resource that can be given to services such as CodePipeline to connect to a third-party repository such as Bitbucket. For example, you can add the connection in CodePipeline so that it triggers your pipeline when a code change is made to your third-party code repository. Each connection is named and associated with a unique ARN that is used to reference the connection.

When you create a connection, the console initiates a third-party connection handshake. **Installations** are the apps that are used to conduct this handshake. For example, the installation for the Bitbucket provider type is the Bitbucket app. When you create a connection, you can choose an existing installation or create one.

When you want to create a connection to an installed provider type such as GitHub Enterprise Server, you create a *host* for your connections.

You can work with connections by calling:

- `create_connection`, which creates a uniquely named connection that can be referenced by services such as CodePipeline.
- `delete_connection`, which deletes the specified connection.
- `get_connection`, which returns information about the connection, including the connection status.
• list_connections, which lists the connections associated with your account.

You can work with hosts by calling:

• create_host, which creates a host that represents the infrastructure where your provider is installed.
• delete_host, which deletes the specified host.
• get_host, which returns information about the host, including the setup status.
• list_hosts, which lists the hosts associated with your account.

You can work with tags in Amazon Web Services CodeStar Connections by calling the following:

• list_tags_for_resource, which gets information about Amazon Web Services tags for a specified Amazon Resource Name (ARN) in Amazon Web Services CodeStar Connections.
• tag_resource, which adds or updates tags for a resource in Amazon Web Services CodeStar Connections.
• untag_resource, which removes tags for a resource in Amazon Web Services CodeStar Connections.

For information about how to use Amazon Web Services CodeStar Connections, see the Developer Tools User Guide.

Usage

codestarconnections(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

codestarconnections Optional configuration of credentials, endpoint, and/or region.

    config  • credentials:
    • credentials:                • endpoint: The complete URL to use for the constructed client.
      • creds:  • region: The AWS Region used in instantiating the client.
        • access_key_id: AWS access key ID
        • secret_access_key: AWS secret access key
        • session_token: AWS temporary session token
        • profile: The name of a profile to use. If not given, then the default profile is used.
        • anonymous: Set anonymous credentials.
    • endpoint: The complete URL to use for the constructed client.
    • region: The AWS Region used in instantiating the client.
    • close_connection: Immediately close all HTTP connections.
    • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.

- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

### credentials

Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.

- **anonymous**: Set anonymous credentials.

### endpoint

Optional shorthand for complete URL to use for the constructed client.

### region

Optional shorthand for AWS Region used in instantiating the client.

### Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)` where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

### Service syntax

```r
svc <- codestarconnections(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    )
  )
)
```
### Operations

- **create_connection**
  - Creates a connection that can then be given to other Amazon Web Services services like CodePipeline so that it can access third-party code repositories.

- **create_host**
  - Creates a resource that represents the infrastructure where a third-party provider is installed.

- **create_repository_link**
  - Creates a link to a specified external Git repository.

- **create_sync_configuration**
  - Creates a sync configuration which allows Amazon Web Services to sync content from a Git repository to update a specified Amazon Web Services resource.

- **delete_connection**
  - Deletes the connection to be deleted.

- **delete_host**
  - Deletes the host to be deleted.

- **delete_repository_link**
  - Deletes the association between your connection and a specified external Git repository.

- **delete_sync_configuration**
  - Deletes the sync configuration for a specified repository and connection.

- **get_connection**
  - Returns the connection ARN and details such as status, owner, and provider type.

- **get_host**
  - Returns the host ARN and details such as status, provider type, endpoint, and, if applicable, the VPC configuration.

- **get_repository_link**
  - Returns details about a repository link.

- **get_repository_sync_status**
  - Returns details about the sync status for a repository.

- **get_resource_sync_status**
  - Returns the status of the sync with the Git repository for a specific Amazon Web Services resource.

- **get_sync_blocker_summary**
  - Returns a list of the most recent sync blockers.

- **get_sync_configuration**
  - Returns details about a sync configuration, including the sync type and resource name.

- **list_connections**
  - Lists the connections associated with your account.

- **list_hosts**
  - Lists the hosts associated with your account.

- **list_repository_links**
  - Lists the repository links created for connections in your account.

- **list_repository_sync_definitions**
  - Lists the repository sync definitions for repository links in your account.

- **list_sync_configurations**
  - Returns a list of sync configurations for a specified repository.

- **list_tags_for_resource**
  - Gets the set of key-value pairs (metadata) that are used to manage the resource.

- **tag_resource**
  - Adds to or modifies the tags of the given resource.

- **untag_resource**
  - Removes tags from an Amazon Web Services resource.

- **update_host**
  - Updates a specified host with the provided configurations.

- **update_repository_link**
  - Updates the association between your connection and a specified external Git repository.

- **update_sync_blocker**
  - Allows you to update the status of a sync blocker, resolving the blocker and allowing syncing to continue.

- **update_sync_configuration**
  - Updates the sync configuration for your connection and a specified external Git repository.

### Examples

```r
## Not run:
svc <- codestarconnections()
svc$create_connection(Foo = 123)

## End(Not run)
```
Description

This AWS CodeStar Notifications API Reference provides descriptions and usage examples of the operations and data types for the AWS CodeStar Notifications API. You can use the AWS CodeStar Notifications API to work with the following objects:

Notification rules, by calling the following:

- `create_notification_rule`, which creates a notification rule for a resource in your account.
- `delete_notification_rule`, which deletes a notification rule.
- `describe_notification_rule`, which provides information about a notification rule.
- `list_notification_rules`, which lists the notification rules associated with your account.
- `update_notification_rule`, which changes the name, events, or targets associated with a notification rule.
- `subscribe`, which subscribes a target to a notification rule.
- `unsubscribe`, which removes a target from a notification rule.

Targets, by calling the following:

- `delete_target`, which removes a notification rule target from a notification rule.
- `list_targets`, which lists the targets associated with a notification rule.

Events, by calling the following:

- `list_event_types`, which lists the event types you can include in a notification rule.

Tags, by calling the following:

- `list_tags_for_resource`, which lists the tags already associated with a notification rule in your account.
- `tag_resource`, which associates a tag you provide with a notification rule in your account.
- `untag_resource`, which removes a tag from a notification rule in your account.

For information about how to use AWS CodeStar Notifications, see the Amazon Web Services Developer Tools Console User Guide.

Usage

```python
codestarnotifications(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```
Arguments

Optional configuration of credentials, endpoint, and/or region.

- **credentials**
  - **creds**:
    * **access_key_id**: AWS access key ID
    * **secret_access_key**: AWS secret access key
    * **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

- **endpoint**: The complete URL to use for the constructed client.

- **region**: The AWS Region used in instantiating the client.

- **close_connection**: Immediately close all HTTP connections.

- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

- **s3_force_path_style**: Set this to **true** to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

**credentials**

Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

**endpoint**

Optional shorthand for complete URL to use for the constructed client.

**region**

Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
csvc <- codestarnotifications(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
      
```
secret_access_key = "string",
session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string",
close_connection = "logical",
timeout = "numeric",
s3_force_path_style = "logical",
sts REGIONAL_ENDPOINT = "string"
),
credentials = list(
creds = list(
access_key_id = "string",
secret_access_key = "string",
session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

create_notification_rule Creates a notification rule for a resource
delete_notification_rule Deletes a notification rule for a resource
delete_target Deletes a specified target for notifications
describe_notification_rule Returns information about a specified notification rule
list_event_types Returns information about the event types available for configuring notifications
list_notification_rules Returns a list of the notification rules for an Amazon Web Services account
list_tags_for_resource Returns a list of the tags associated with a notification rule
list_targets Returns a list of the notification rule targets for an Amazon Web Services account
subscribe Creates an association between a notification rule and an Chatbot topic or Chatbot client so that the
                    associated target can receive notifications when the events described in the rule are triggered
tag_resource Associates a set of provided tags with a notification rule
unsubscribe Removes an association between a notification rule and an Chatbot topic so that subscribers to that
                    topic stop receiving notifications when the events described in the rule are triggered
untag_resource Removes the association between one or more provided tags and a notification rule
update_notification_rule Updates a notification rule for a resource

Examples

## Not run:
svc <- codestarnotifications()
cognitoidentity

description

Amazon Cognito Federated Identities

Amazon Cognito Federated Identities is a web service that delivers scoped temporary credentials to mobile devices and other untrusted environments. It uniquely identifies a device and supplies the user with a consistent identity over the lifetime of an application.

Using Amazon Cognito Federated Identities, you can enable authentication with one or more third-party identity providers (Facebook, Google, or Login with Amazon) or an Amazon Cognito user pool, and you can also choose to support unauthenticated access from your app. Cognito delivers a unique identifier for each user and acts as an OpenID token provider trusted by AWS Security Token Service (STS) to access temporary, limited-privilege AWS credentials.

For a description of the authentication flow from the Amazon Cognito Developer Guide see Authentication Flow.

For more information see Amazon Cognito Federated Identities.

Usage

cognitoidentity(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)

Arguments

config Optional configuration of credentials, endpoint, and/or region.

  credentials:
    - creds:
      - access_key_id: AWS access key ID
      - secret_access_key: AWS secret access key
      - session_token: AWS temporary session token
    - profile: The name of a profile to use. If not given, then the default profile is used.
    - anonymous: Set anonymous credentials.
• **endpoint**: The complete URL to use for the constructed client.
• **region**: The AWS Region used in instantiating the client.
• **close_connection**: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

**credentials**

Optional credentials shorthand for the config parameter

• **creds**:
  – **access_key_id**: AWS access key ID
  – **secret_access_key**: AWS secret access key
  – **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.
• **anonymous**: Set anonymous credentials.

**endpoint**

Optional shorthand for complete URL to use for the constructed client.

**region**

Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- cognitoidentity(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  )
)
```
cognitoidentity

),
credentials = list(
cred = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

create_identity_pool
create_identity_pool
delete_identities
delete_identity_pool
describe_identity
describe_identity_pool
get_credentials_for_identity
get_id
get_identity_pool_roles
get_open_id_token
get_open_id_token_for_developer_identity
get_principal_tag_attribute_map
list_identities
list_identity_pools
list_tags_for_resource
lookup_developer_identity
merge_developer_identities
set_identity_pool_roles
set_principal_tag_attribute_map
tag_resource
unlink_developer_identity
unlink_identity
untag_resource
update_identity_pool

Creates a new identity pool
Deletes identities from an identity pool
Deletes an identity pool
Returns metadata related to the given identity, including when the identity was created
Gets details about a particular identity pool, including the pool name, ID description, creation date, and current number of users
Generates (or retrieves) a Cognito ID
Gets the roles for an identity pool
Gets an OpenID token, using a known Cognito ID
Registers (or retrieves) a Cognito IdentityId and an OpenID Connect token for a user
Use GetPrincipalTagAttributeMap to list all mappings between PrincipalTags and user attributes
Lists the identities in an identity pool
Lists all of the Cognito identity pools registered for your account
Lists the tags that are assigned to an Amazon Cognito identity pool
Retrieves the IdentityID associated with a DeveloperUserIdentifier or the list of
Merges two users having different IdentityIds, existing in the same identity pool
Sets the roles for an identity pool
You can use this operation to use default (username and clientID) attribute or custom
Assigns a set of tags to the specified Amazon Cognito identity pool
Unlinks a DeveloperUserIdentifier from an existing identity
Unlinks a federated identity from an existing account
Removes the specified tags from the specified Amazon Cognito identity pool
Updates an identity pool

Examples

## Not run:
svc <- cognitoidentity()
svc$create_identity_pool(
    Foo = 123
)
With the Amazon Cognito user pools API, you can configure user pools and authenticate users. To authenticate users from third-party identity providers (IdPs) in this API, you can link IdP users to native user profiles. Learn more about the authentication and authorization of federated users at Adding user pool sign-in through a third party and in the User pool federation endpoints and hosted UI reference.

This API reference provides detailed information about API operations and object types in Amazon Cognito. Along with resource management operations, the Amazon Cognito user pools API includes classes of operations and authorization models for client-side and server-side authentication of users. You can interact with operations in the Amazon Cognito user pools API as any of the following subjects.

1. An administrator who wants to configure user pools, app clients, users, groups, or other user pool functions.
2. A server-side app, like a web application, that wants to use its Amazon Web Services privileges to manage, authenticate, or authorize a user.
3. A client-side app, like a mobile app, that wants to make unauthenticated requests to manage, authenticate, or authorize a user.

For more information, see Using the Amazon Cognito user pools API and user pool endpoints in the Amazon Cognito Developer Guide.

With your Amazon Web Services SDK, you can build the logic to support operational flows in every use case for this API. You can also make direct REST API requests to Amazon Cognito user pools service endpoints. The following links can get you started with the CognitoIdentityProvider client in other supported Amazon Web Services SDKs.

- Amazon Web Services Command Line Interface
- Amazon Web Services SDK for .NET
- Amazon Web Services SDK for C++
- Amazon Web Services SDK for Go
- Amazon Web Services SDK for Java V2
- Amazon Web Services SDK for JavaScript
- Amazon Web Services SDK for PHP V3
- Amazon Web Services SDK for Python
• Amazon Web Services SDK for Ruby V3

To get started with an Amazon Web Services SDK, see Tools to Build on Amazon Web Services. For example actions and scenarios, see Code examples for Amazon Cognito Identity Provider using Amazon Web Services SDKs.

Usage

cognitoidentityprovider(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

cconfig Optional configuration of credentials, endpoint, and/or region.
    • credentials:
        – creds:
            * access_key_id: AWS access key ID
            * secret_access_key: AWS secret access key
            * session_token: AWS temporary session token
        – profile: The name of a profile to use. If not given, then the default profile is used.
        – anonymous: Set anonymous credentials.
    • endpoint: The complete URL to use for the constructed client.
    • region: The AWS Region used in instantiating the client.
    • close_connection: Immediately close all HTTP connections.
    • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
    • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
    • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter
    • creds:
        – access_key_id: AWS access key ID
        – secret_access_key: AWS secret access key
        – session_token: AWS temporary session token
        • profile: The name of a profile to use. If not given, then the default profile is used.
        • anonymous: Set anonymous credentials.

cendpoint Optional shorthand for complete URL to use for the constructed client.

cregion Optional shorthand for AWS Region used in instantiating the client.
Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```
svc <- cognitoidentityprovider(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- **add_custom_attributes**: Adds additional user attributes to the user pool schema
- **admin_add_user_to_group**: Adds a user to a group
- **admin_confirm_sign_up**: This IAM-authenticated API operation provides a code that Amazon Cognito sent to your user when they signed up in your user pool
- **admin_create_user**: Creates a new user in the specified user pool
- **admin_delete_user**: Deletes a user as an administrator
- **admin_delete_user_attributes**: Deletes the user attributes in a user pool as an administrator
- **admin_disable_provider_for_user**: Prevents the user from signing in with the specified external (SAML or social) identity provider
- **admin_disable_user**: Deactivates a user and revokes all access tokens for the user
admin_enable_user
admin_forget_device
admin_get_device
admin_get_user
admin_initiate_auth
admin_link_provider_for_user
admin_list_devices
admin_list_groups_for_user
admin_list_user_auth_events
admin_remove_user_from_group
admin_reset_user_password
admin_respond_to_auth_challenge
admin_set_user_mfa_preference
admin_set_user_password
admin_set_user_settings
admin_update_auth_event_feedback
admin_update_device_status
admin_update_user_attributes
admin_user_global_sign_out
associate_software_token
change_password
confirm_device
confirm_forgot_password
confirm_sign_up
create_group
create_identity_provider
create_resource_server
create_user_import_job
create_user_pool
create_user_pool_client
create_user_pool_domain
delete_group
delete_identity_provider
delete_resource_server
delete_user
delete_user_attributes
delete_user_pool
delete_user_pool_client
delete_user_pool_domain
describe_identity_provider
describe_resource_server
describe_risk_configuration
describe_user_import_job
describe_user_pool
describe_user_pool_client
describe_user_pool_domain
forget_device
forgot_password

Enables the specified user as an administrator
Forgets the device, as an administrator
Gets the device, as an administrator
Gets the specified user by user name in a user pool as an administrator
Initiates the authentication flow, as an administrator
Links an existing user account in a user pool (DestinationUser) to an identity from an external IdP
Lists devices, as an administrator
Lists the groups that a user belongs to
A history of user activity and any risks detected as part of Amazon Cognito advanced security
Removes the specified user from the specified group
Resets the specified user’s password in a user pool as an administrator
Some API operations in a user pool generate a challenge, like a prompt for an MFA code
The user’s multi-factor authentication (MFA) preference, including which MFA options are activated
Sets the specified user’s password in a user pool as an administrator
This action is no longer supported
Provides feedback for an authentication event indicating if it was from a valid user
Updates the device status as an administrator
This action might generate an SMS text message
Invalidates the identity, access, and refresh tokens that Amazon Cognito issued to a user
Begins setup of time-based one-time password (TOTP) multi-factor authentication (MFA)
Changes the password for a specified user in a user pool
Confirms tracking of the device
Allows a user to enter a confirmation code to reset a forgotten password
This public API operation provides a code that Amazon Cognito sent to your user when they signed up
Creates a new group in the specified user pool
Adds a configuration and trust relationship between a third-party identity provider (IdP)
Creates a new OAuth2
Creates a user import job
This action might generate an SMS text message
Creates the user pool client
Creates a new domain for a user pool
Deletes a group
Deletes an IdP for a user pool
Deletes a resource server
Allows a user to delete their own user profile
Deletes the attributes for a user
Deletes the specified Amazon Cognito user pool
Allows the developer to delete the user pool client
Deletes a domain for a user pool
Gets information about a specific IdP
Describes a resource server
Describes the risk configuration
Describes the user import job
Returns the configuration information and metadata of the specified user pool
Client method for returning the configuration information and metadata of the specified user pool
Gets information about a domain
Forgets the specified device
Calling this API causes a message to be sent to the end user with a confirmation code th
get_csv_header
get_device
get_group
get_identity_provider_by_identifier
get_log_delivery_configuration
get_signing_certificate
get_ui_customization
get_user
get_user_attribute_verification_code
get_user_pool_mfa_config
global_sign_out
initiate_auth
list_devices
list_groups
list_identity_providers
list_resource_servers
list_tags_for_resource
list_user_import_jobs
list_user_pool_clients
list_user_pools
list_users
list_users_in_group
resend_confirmation_code
respond_to_auth_challenge
revoke_token
set_log_delivery_configuration
set_risk_configuration
set_ui_customization
set_user_mfa_preference
set_user_pool_mfa_config
set_user_settings
sign_up
start_user_import_job
stop_user_import_job
tag_resource
untag_resource
update_auth_event_feedback
update_device_status
update_group
update_identity_provider
update_resource_server
update_user_attributes
update_user_pool
update_user_pool_client
update_user_pool_domain
verify_software_token
verify_user_attribute

get_csv_header Gets the header information for the comma-separated value (CSV) file to be used as input for the user import job
get_device Gets the device
get_group Gets a group
get_identity_provider_by_identifier Gets the specified IdP
get_log_delivery_configuration Gets the detailed activity logging configuration for a user pool
get_signing_certificate This method takes a user pool ID, and returns the signing certificate
get_ui_customization Gets the user interface (UI) Customization information for a particular app client’s app UI
get_user Gets the user attributes and metadata for a user
get_user_attribute_verification_code Generates a user attribute verification code for the specified attribute name
get_user_pool_mfa_config Gets the user pool multi-factor authentication (MFA) configuration
get_user_pool_mfa_config
resend_confirmation_code
respond_to_auth_challenge
revoke_token
set_log_delivery_configuration
set_risk_configuration
set_ui_customization
set_user_mfa_preference
set_user_pool_mfa_config
set_user_settings
sign_up
start_user_import_job
stop_user_import_job
tag_resource
untag_resource
update_auth_event_feedback
update_device_status
update_group
update_identity_provider
update_resource_server
update_user_attributes
update_user_pool
update_user_pool_client
update_user_pool_domain
verify_software_token
verify_user_attribute

Invalidates the identity, access, and refresh tokens that Amazon Cognito issued to a user
Initiates sign-in for a user in the Amazon Cognito user directory
Lists the sign-in devices that Amazon Cognito has registered to the current user
Lists the groups associated with a user pool
Lists information about all IdPs for a user pool
Lists the resource servers for a user pool
Lists the groups that are assigned to an Amazon Cognito user pool
Lists user import jobs for a user pool
Lists the clients that have been created for the specified user pool
Lists the users associated with an Amazon Web Services account
Lists users and their basic details in a user pool
Lists the users in the specified group
Resends the confirmation (for confirmation of registration) to a specific user in the user pool
Some API operations in a user pool generate a challenge, like a prompt for an MFA code
Rerevokes all of the access tokens generated by, and at the same time as, the specified refresh token
Sets up or modifies the detailed activity logging configuration of a user pool
Configures actions on detected risks
Sets the user interface (UI) customization information for a user pool’s built-in app UI
Set the user’s multi-factor authentication (MFA) method preference, including which MFA factors are activated and if any are preferred
Sets the user pool multi-factor authentication (MFA) configuration
This action is no longer supported
Registers the user in the specified user pool and creates a user name, password, and user attributes
Starts the user import
Stops the user import job
Assigns a set of tags to an Amazon Cognito user pool
Removes the specified tags from an Amazon Cognito user pool
Provides the feedback for an authentication event, whether it was from a valid user or not
Updates the device status
Updates the specified group with the specified attributes
Updates IdP information for a user pool
Updates the name and scopes of resource server
With this operation, your users can update one or more of their attributes with their own credentials
This action might generate an SMS text message
Updates the specified user pool app client with the specified attributes
Updates the Secure Sockets Layer (SSL) certificate for the custom domain for your user pool
Use this API to register a user’s entered time-based one-time password (TOTP) code and mark their software token MFA status as "verified" if successful
Verifies the specified user attributes in the user pool
Examples

```r
## Not run:
svc <- cognitoidentityprovider()
# This request submits a value for all possible parameters for
# AdminCreateUser.
svc$admin_create_user(
  DesiredDeliveryMediums = list(“SMS”),
  MessageAction = “SUPPRESS”,
  TemporaryPassword = “This-is-my-test-99!”,
  UserAttributes = list(  
    list(  
      Name = “name”,
      Value = “John”
    ),
    list(  
      Name = “phone_number”,
      Value = “+12065551212”
    ),
    list(  
      Name = “email”,
      Value = “testuser@example.com”
    )  
  ),
  UserPoolId = “us-east-1_EXAMPLE”,
  Username = “testuser”
)

## End(Not run)
```

Amazon Cognito Sync

Amazon Cognito Sync provides an AWS service and client library that enable cross-device syncing of application-related user data. High-level client libraries are available for both iOS and Android. You can use these libraries to persist data locally so that it’s available even if the device is offline. Developer credentials don’t need to be stored on the mobile device to access the service. You can use Amazon Cognito to obtain a normalized user ID and credentials. User data is persisted in a dataset that can store up to 1 MB of key-value pairs, and you can have up to 20 datasets per user identity.

With Amazon Cognito Sync, the data stored for each identity is accessible only to credentials assigned to that identity. In order to use the Cognito Sync service, you need to make API calls using credentials retrieved with Amazon Cognito Identity service.
If you want to use Cognito Sync in an Android or iOS application, you will probably want to make API calls via the AWS Mobile SDK. To learn more, see the Developer Guide for Android and the Developer Guide for iOS.

Usage

cognitosync(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

config  Optional configuration of credentials, endpoint, and/or region.
    • credentials:
      – creds:
        * access_key_id: AWS access key ID
        * secret_access_key: AWS secret access key
        * session_token: AWS temporary session token
      – profile: The name of a profile to use. If not given, then the default profile is used.
      – anonymous: Set anonymous credentials.
    • endpoint: The complete URL to use for the constructed client.
    • region: The AWS Region used in instantiating the client.
    • close_connection: Immediately close all HTTP connections.
    • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
    • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
    • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials  Optional credentials shorthand for the config parameter
    • creds:
      – access_key_id: AWS access key ID
      – secret_access_key: AWS secret access key
      – session_token: AWS temporary session token
    • profile: The name of a profile to use. If not given, then the default profile is used.
    • anonymous: Set anonymous credentials.

defaults

endpoint  Optional shorthand for complete URL to use for the constructed client.
region  Optional shorthand for AWS Region used in instantiating the client.
Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- cognitosync(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- **bulk_publish**
  - Initiates a bulk publish of all existing datasets for an Identity Pool to the configured stream
- **delete_dataset**
  - Deletes the specific dataset
- **describe_dataset**
  - Gets meta data about a dataset by identity and dataset name
- **describe_identity_pool_usage**
  - Gets usage details (for example, data storage) about a particular identity pool
- **describe_identity_usage**
  - Gets usage information for an identity, including number of datasets and data usage
- **get_bulk_publish_details**
  - Get the status of the last BulkPublish operation for an identity pool
- **get_cognito_events**
  - Gets the events and the corresponding Lambda functions associated with an identity pool
- **get_identity_pool_configuration**
  - Gets the configuration settings of an identity pool
### Comprehend

**list_datasets**
- Lists datasets for an identity

**list_identity_pool_usage**
- Gets a list of identity pools registered with Cognito

**list_records**
- Gets paginated records, optionally changed after a particular sync count for a dataset and identity

**register_device**
- Registers a device to receive push sync notifications

**set_cognito_events**
- Sets the AWS Lambda function for a given event type for an identity pool

**set_identity_pool_configuration**
- Sets the necessary configuration for push sync

**subscribe_to_dataset**
- Subscribes to receive notifications when a dataset is modified by another device

**unsubscribe_from_dataset**
- Unsubscribes from receiving notifications when a dataset is modified by another device

**update_records**
- Posts updates to records and adds and deletes records for a dataset and user

---

**Examples**

```r
## Not run:
svc <- cognitosync()
svc$bulk_publish(
  Foo = 123
)

## End(Not run)
```

### Description

Amazon Comprehend is an Amazon Web Services service for gaining insight into the content of documents. Use these actions to determine the topics contained in your documents, the topics they discuss, the predominant sentiment expressed in them, the predominant language used, and more.

### Usage

```r
comprehend(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

### Arguments

- **config**
  - Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
* secret_access_key: AWS secret access key
* session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.

• endpoint: The complete URL to use for the constructed client.
• region: The AWS Region used in instantiating the client.
• close_connection: Immediately close all HTTP connections.
• timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    - access_key_id: AWS access key ID
    - secret_access_key: AWS secret access key
    - session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.
region Optional shorthand for AWS Region used in instantiating the client.

Value
A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax
```r
csvc <- comprehend(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
```
region = "string",
close_connection = "logical",
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

`batch_detect_dominant_language` Determines the dominant language of the input text for a batch of documents

`batch_detect_entities` Inspects the text of a batch of documents for named entities and returns information about them

`batch_detect_key_phrases` Detects the key noun phrases found in a batch of documents

`batch_detect_sentiment` Inspects a batch of documents and returns an inference of the prevailing sentiment, POSITIVE, NEUTRAL, MIXED, or NEGATIVE, in each one

`batch_detect_syntax` Inspects the text of a batch of documents for the syntax and part of speech of the words in the document and returns information about them

`batch_detect_targeted_sentiment` Inspects a batch of documents and returns a sentiment analysis for each entity identified in the documents

`classify_document` Creates a classification request to analyze a single document in real-time

`contains_pii_entities` Analyzes input text for the presence of personally identifiable information (PII) and returns the labels of identified PII entity types such as name, address, bank account number, or phone number

`create_dataset` Creates a dataset to upload training or test data for a model associated with a flywheel

`create_document_classifier` Creates a new document classifier that you can use to categorize documents

`create_endpoint` Creates a model-specific endpoint for synchronous inference for a previously trained custom model

`create_entity_recognizer` Creates an entity recognizer using submitted files

`create_flywheel` A flywheel is an Amazon Web Services resource that orchestrates the ongoing training of a model for custom classification or custom entity recognition

`delete_document_classifier` Deletes a previously created document classifier

`delete_endpoint` Deletes a model-specific endpoint for a previously-trained custom model

`delete_entity_recognizer` Deletes an entity recognizer

`delete_flywheel` Deletes a flywheel

`delete_resource_policy` Deletes a resource-based policy that is attached to a custom model

`describe_dataset` Returns information about the dataset that you specify

`describe_document_classification_job` Gets the properties associated with a document classification job

`describe_document_classifier` Gets the properties associated with a document classifier

`describe_dominant_language_detection_job` Gets the properties associated with a dominant language detection job

`describe_endpoint` Gets the properties associated with a specific endpoint

`describe_entities_detection_job` Provides details about an entity recognizer including status, S3 buckets containing training data, recognizer metadata, metrics, and so on

`describe_entity_recognizer` Gets the status and details of an events detection job

`describe_events_detection_job` Provides configuration information about the flywheel

`describe_flywheel`
describe_flywheel_iteration
describe_key_phrases_detection_job
describe_pii_entities_detection_job
describe_resource_policy
describe_sentiment_detection_job
describe_targeted_sentiment_detection_job
describe_topics_detection_job
detect_dominant_language
detect_entities
detect_key_phrases
detect_pii_entities
detect_sentiment
detect_syntax
detect_targeted_sentiment
detect_toxic_content
import_model
list_datasets
list_document_classification_jobs
list_document_classifiers
list_document_classifier_summaries
list_dominant_language_detection_jobs
list_endpoints
list_entities_detection_jobs
list_entity_recognizers
list_entity_recognizer_summaries
list_events_detection_jobs
list_flywheel_iteration_history
list_flywheels
list_key_phrases_detection_jobs
list_pii_entities_detection_jobs
list_sentiment_detection_jobs
list_tags_for_resource
list_targeted_sentiment_detection_jobs
list_topics_detection_jobs
put_resource_policy
start_document_classification_job
start_dominant_language_detection_job
start_entities_detection_job
start_events_detection_job
start_flywheel_iteration
start_key_phrases_detection_job
start_pii_entities_detection_job
start_sentiment_detection_job
start_targeted_sentiment_detection_job
start_topics_detection_job
stop_dominant_language_detection_job
stop_entities_detection_job
stop_events_detection_job

Retrieve the configuration properties of a flywheel iteration
Gets the properties associated with a key phrases detection job
Gets the properties associated with a PII entities detection job
Gets the details of a resource-based policy that is attached to a custom model, in JSON format
Gets the properties associated with a sentiment detection job
Gets the properties associated with a targeted sentiment detection job
Determines the dominant language of the input text
Detects named entities in input text when you use the pre-trained model
Inspects the input text for entities that contain personally identifiable information (PII)
Inspects text and returns an inference of the prevailing sentiment (POSITIVE, NEGATIVE, MIXED, or NEUTRAL)
Inspects the input text and returns a sentiment analysis for each entity identified in the text
Performs toxicity analysis on the list of text strings that you provide as input
Creates a new custom model that replicates a source custom model that you import
List the datasets that you have configured in this Region
Gets a list of the documentation classification jobs that you have submitted
Gets a list of the document classifiers that you have created
Gets a list of summaries of the document classifiers that you have created
Gets a list of the dominant language detection jobs that you have submitted
Gets a list of all existing endpoints that you’ve created
Gets a list of the entity detection jobs that you have submitted
Gets a list of the properties of all entity recognizers that you created, including recognizers currently in training
Gets a list of summaries for the entity recognizers that you have created
Gets a list of the events detection jobs that you have submitted
Information about the history of a flywheel iteration
Gets a list of the flywheels that you have created
Get a list of key phrase detection jobs that you have submitted
Gets a list of the PII entity detection jobs that you have submitted
Gets a list of sentiment detection jobs that you have submitted
Lists all tags associated with a given Amazon Comprehend resource
Gets a list of targeted sentiment detection jobs that you have submitted
Gets a list of the topic detection jobs that you have submitted
Attaches a resource-based policy to a custom model
Starts an asynchronous document classification job using a custom classification model
Starts an asynchronous dominant language detection job for a collection of documents
Starts an asynchronous entity detection job for a collection of documents
Starts an asynchronous event detection job for a collection of documents
Start the flywheel iteration
Starts an asynchronous key phrase detection job for a collection of documents
Starts an asynchronous PII entity detection job for a collection of documents
Starts an asynchronous sentiment detection job for a collection of documents
Starts an asynchronous targeted sentiment detection job for a collection of documents
Starts an asynchronous topic detection job
Stops a dominant language detection job in progress
Stops an entities detection job in progress
Stops an events detection job in progress
### Examples

```r
## Not run:
svc <- comprehend()
svc$batch_detect_dominant_language(
  Foo = 123
)
## End(Not run)
```

---

**comprehendmedical**  
**AWS Comprehend Medical**

#### Description

Amazon Comprehend Medical extracts structured information from unstructured clinical text. Use these actions to gain insight in your documents. Amazon Comprehend Medical only detects entities in English language texts. Amazon Comprehend Medical places limits on the sizes of files allowed for different API operations. To learn more, see Guidelines and quotas in the Amazon Comprehend Medical Developer Guide.

#### Usage

```r
comprehendmedical(  
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```
Arguments

config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default
      profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when at-
    tempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style
    addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile
    is used.
  • anonymous: Set anonymous credentials.

describe

endpoint Optional shorthand for complete URL to use for the constructed client.
region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...),
where svc is the name you’ve assigned to the client. The available operations are listed in the Op-
erations section.

Service syntax

svc <- comprehendmedical(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
      )
    )
  )
)
Operations

describe_entities_detection_v2_job
    Gets the properties associated with a medical entities detection job

describe_icd10cm_inference_job
    Gets the properties associated with an InferICD10CM job

describe_phi_detection_job
    Gets the properties associated with a protected health information (PHI) detection job

describe_rx_norm_inference_job
    Gets the properties associated with an InferRxNorm job

describe_snomedct_inference_job
    Gets the properties associated with an InferSNOMEDCT job

detect_entities_v2
    The DetectEntities operation is deprecated

detect_entities
    Inspects the clinical text for a variety of medical entities and returns specific information about them such as entity category, location, and confidence score on that information

detect_phi
    Inspects the clinical text for protected health information (PHI) entities and returns the entity category, location, and confidence score for each entity

infer_icd10cm
    InferICD10CM detects medical conditions as entities listed in a patient record and links those entities to normalized concept identifiers in the ICD-10-CM knowledge base from the Centers for Disease Control

infer_rx_norm
    InferRxNorm detects medications as entities listed in a patient record and links them to the normalized concept identifiers in the RxNorm database from the National Library of Medicine

infer_snomedct
    InferSNOMEDCT detects possible medical concepts as entities and links them to codes in the Systematized Nomenclature of Medicine, Clinical Terms (SNOMED-CT) ontology

list_entities_detection_v2_jobs
    Gets a list of medical entity detection jobs that you have submitted

list_icd10cm_inference_jobs
    Gets a list of InferICD10CM jobs that you have submitted

list_phi_detection_jobs
    Gets a list of protected health information (PHI) detection jobs you have submitted

list_rx_norm_inference_jobs
    Gets a list of InferRxNorm jobs that you have submitted

list_snomedct_inference_jobs
    Gets a list of InferSNOMEDCT jobs a user has submitted

start_entities_detection_v2_job
    Starts an asynchronous medical entity detection job for a collection of documents

start_icd10cm_inference_job
    Starts an asynchronous job to detect medical conditions and link them to the ICD-10-CM ontology

start_phi_detection_job
    Starts an asynchronous job to detect protected health information (PHI)

start_rx_norm_inference_job
    Starts an asynchronous job to detect medication entities and link them to the RxNorm ontology

secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
),
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
computeoptimizer  

### start_snomedct_inference_job
Starts an asynchronous job to detect medical concepts and link them to the SNOMED-CT ontology

### stop_entities_detection_v2_job
Stops a medical entities detection job in progress

### stop_icd10cm_inference_job
Stops an InferICD10CM inference job in progress

### stop_phi_detection_job
Stops a protected health information (PHI) detection job in progress

### stop_rx_norm_inference_job
Stops an InferRxNorm inference job in progress

### stop_snomedct_inference_job
Stops an InferSNOMEDCT inference job in progress

---

**Examples**

```r
## Not run:
svc <- comprehendmedical()
svc$describe_entities_detection_v2_job(
  Foo = 123
)
## End(Not run)
```

---

**computeoptimizer**  

*AWS Compute Optimizer*

---

**Description**

Compute Optimizer is a service that analyzes the configuration and utilization metrics of your Amazon Web Services compute resources, such as Amazon EC2 instances, Amazon EC2 Auto Scaling groups, Lambda functions, Amazon EBS volumes, and Amazon ECS services on Fargate. It reports whether your resources are optimal, and generates optimization recommendations to reduce the cost and improve the performance of your workloads. Compute Optimizer also provides recent utilization metric data, in addition to projected utilization metric data for the recommendations, which you can use to evaluate which recommendation provides the best price-performance trade-off. The analysis of your usage patterns can help you decide when to move or resize your running resources, and still meet your performance and capacity requirements. For more information about Compute Optimizer, including the required permissions to use the service, see the Compute Optimizer User Guide.

---

**Usage**

```r
computeoptimizer(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```
Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>config</td>
<td>Optional configuration of credentials, endpoint, and/or region.</td>
</tr>
<tr>
<td>credentials</td>
<td>Optional credentials shorthand for the config parameter</td>
</tr>
<tr>
<td>endpoint</td>
<td>Optional shorthand for complete URL to use for the constructed client.</td>
</tr>
<tr>
<td>region</td>
<td>Optional shorthand for AWS Region used in instantiating the client.</td>
</tr>
<tr>
<td>close_connection</td>
<td>Immediately close all HTTP connections.</td>
</tr>
<tr>
<td>timeout</td>
<td>The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.</td>
</tr>
<tr>
<td>s3_force_path_style</td>
<td>Set this to true to force the request to use path-style addressing, i.e. <a href="http://s3.amazonaws.com/BUCKET/KEY">http://s3.amazonaws.com/BUCKET/KEY</a>.</td>
</tr>
<tr>
<td>sts_regional_endpoint</td>
<td>Set sts regional endpoint resolver to regional or legacy <a href="https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html">https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html</a></td>
</tr>
</tbody>
</table>

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- computeoptimizer(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string",
      ),
      profile = "string",
      anonymous = TRUE,
    ),
    endpoint = "string",
    region = "string",
    close_connection = TRUE,
    timeout = 60L,
    s3_force_path_style = TRUE,
    sts_regional_endpoint = "string",
  ),
)```
Operations

delete_recommendation_preferences
describe_recommendation_export_jobs
export_auto_scaling_group_recommendations
export_ebs_volume_recommendations
export_ec2_instance_recommendations
export_ecs_service_recommendations
export_lambda_function_recommendations
export_license_recommendations
get_auto_scaling_group_recommendations
get_ebs_volume_recommendations
get_ec2_instance_recommendations
get_ec2_recommendation_projected_metrics
get_ecs_service_recommendation_projected_metrics
get_ecs_service_recommendations
get_effective_recommendation_preferences
get_enrollment_status
get_enrollment_statuses_for_organization
get_lambda_function_recommendations
get_license_recommendations
get_recommendation_preferences

Deletes a recommendation preference, such as enhanced infrastructure metrics
Describes recommendation export jobs created in the last seven days
Exports optimization recommendations for Auto Scaling groups
Exports optimization recommendations for Amazon EBS volumes
Exports optimization recommendations for Amazon EC2 instances
Exports optimization recommendations for Amazon ECS services on Fargate
Exports optimization recommendations for Lambda functions
Export optimization recommendations for your licenses
Returns Auto Scaling group recommendations
Returns Amazon Elastic Block Store (Amazon EBS) volume recommendations
Returns Amazon EC2 instance recommendations
Returns the projected utilization metrics of Amazon EC2 instance recommendations
Returns the projected metrics of Amazon ECS service recommendations
Returns Amazon ECS service recommendations
Returns the recommendation preferences that are in effect for a given resource
Returns the enrollment (opt in) status of an account to the Compute Optimizer service
Returns the Compute Optimizer enrollment (opt-in) status of organization members
Returns Lambda function recommendations
Returns license recommendations for Amazon EC2 instances that run on
Returns existing recommendation preferences, such as enhanced infrastructure metrics
**Description**

Config

Config provides a way to keep track of the configurations of all the Amazon Web Services resources associated with your Amazon Web Services account. You can use Config to get the current and historical configurations of each Amazon Web Services resource and also to get information about the relationship between the resources. An Amazon Web Services resource can be an Amazon Compute Cloud (Amazon EC2) instance, an Elastic Block Store (EBS) volume, an elastic network Interface (ENI), or a security group. For a complete list of resources currently supported by Config, see Supported Amazon Web Services resources.

You can access and manage Config through the Amazon Web Services Management Console, the Amazon Web Services Command Line Interface (Amazon Web Services CLI), the Config API, or the Amazon Web Services SDKs for Config. This reference guide contains documentation for the Config API and the Amazon Web Services CLI commands that you can use to manage Config. The Config API uses the Signature Version 4 protocol for signing requests. For more information about how to sign a request with this protocol, see Signature Version 4 Signing Process. For detailed information about Config features and their associated actions or commands, as well as how to work with Amazon Web Services Management Console, see What Is Config in the Config Developer Guide.

**Usage**

```r
configs service(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

**Examples**

```r
## Not run:
svc <- computeoptimizer()
svc$delete_recommendation_preferences(
   Foo = 123
)
## End(Not run)
```
Arguments

config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```
svc <- configservice(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string",
      ),
      profile = "string",
      anonymous = FALSE
    ),
    endpoint = "string",
    region = "string",
    close_connection = FALSE,
    timeout = 60,
    s3_force_path_style = FALSE,
    sts_regional_endpoint = "string"
  ),
  
```
secret_access_key = "string",
session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string",
close_connection = "logical",
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
creds = list(
  access_key_id = "string",
  secret_access_key = "string",
  session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

batch_get_aggregate_resource_config
batch_get_resource_config
delete_aggregation_authorization
delete_config_rule
delete_configuration_aggregator
delete_configuration_recorder
delete_conformance_pack
delete_delivery_channel
delete_evaluation_results
delete_organization_config_rule
delete_organization_conformance_pack
delete_pending_aggregation_request
delete_remediation_configuration
delete_remediation_exceptions
delete_resource_config
delete_retention_configuration
delete_stored_query
deliver_config_snapshot
describe_aggregate_compliance_by_config_rules
describe_aggregate_compliance_by_conformance_packs

Returns the current configuration items for resources that are present in your Config aggregator.
Returns the BaseConfigurationItem for one or more requested resources.
Deletes the authorization granted to the specified configuration aggregator account.
Deletes the specified Config rule and all of its evaluation results.
Deletes the specified configuration aggregator and the aggregated data associated with it.
Deletes the configuration recorder.
Deletes the specified conformance pack and all of the Config rules, remediation actions, and all evaluation results associated with it.
Deletes the delivery channel.
Deletes the evaluation results for the specified Config rule.
Deletes the specified organization Config rule and all of its evaluation results.
Deletes the specified organization conformance pack and all of the evaluation results associated with it.
Deletes pending authorization requests for a specified aggregator account.
Deletes the remediation configuration.
Deletes one or more remediation exceptions mentioned in the resource key.
Records the configuration state for a custom resource that has been deleted.
Deletes the retention configuration.
Deletes the stored query for a single Amazon Web Services account.
Schedules delivery of a configuration snapshot to the Amazon S3.
Returns a list of compliant and noncompliant rules with the number of resources for each rule.
Returns a list of the conformance packs and their associated compliance status.
describe_aggregation_authorizations
describe_compliance_by_config_rule
describe_compliance_by_resource
describe_config_rule_evaluation_status
describe_config_rules
describe_configuration_aggregators
describe_configuration_aggregator_sources_status
describe_configuration_recorders
describe_configuration_recorder_status
describe_conformance_pack_compliance
describe_conformance_packs
describe_delivery_channels
describe_delivery_channel_status
describe_organization_config_rules
describe_organization_config_rule_statuses
describe_organization_conformance_packs
describe_organization_conformance_pack_statuses
describe_pending_aggregation_requests
describe_remediation_configurations
describe_remediation_exceptions
describe_remediation_execution_status
describe_retention_configurations
describe_aggregate_compliance_details_by_config_rule
describe_aggregate_conformance_pack_compliance_summary
describe_aggregate_discovered_resource_counts
describe_aggregate_discovered_resources
describe_aggregate_discovered_resource_counts
get_aggregate_discovered_resource_counts
get_custom_rule_policy
get_discovered_resource_counts
get_organization_config_rule_detailed_status
get_organization_conformance_pack_detailed_status
get_resource_config
get_resource_config_history
get_resource_evaluation_summary
get_stored_query
list_aggregate_discovered_resources
list_conformance_pack_compliance_scores
list_discovered_resources
list_resource_evaluations
list_stored_queries
list_tags_for_resource

Returns a list of authorizations granted to various aggregator accounts.
Indicates whether the specified Config rules are compliant.
Indicates whether the specified Amazon Web Services resources are compliant.
Returns status information for each of your Config managed rules.
Returns details about your Config rules.
Returns the details of one or more configuration aggregators.
Returns status information for sources within an aggregator.
Returns the details for the specified configuration recorders.
Returns the current status of the specified configuration recorder.
Returns compliance details for each rule in that conformance pack.
Returns a list of one or more conformance packs.
Provides one or more conformance packs deployment status.
Returns details about the specified delivery channel.
Returns the current status of the specified delivery channel.
Returns a list of organization Config rules.
Provides organization Config rule deployment status for an organization.
Returns a list of organization conformance packs.
Provides organization conformance pack deployment status for an organization.
Returns a list of all pending aggregation requests.
Returns the details of one or more remediation configurations.
Returns the details of one or more remediation exceptions.
Provides a detailed view of a Remediation Execution for a set of resources.
Returns the details of one or more retention configurations.
Returns the evaluation results for the specified Config rule for a set of resources.
Returns the number of compliant and noncompliant rules for one Config rule.
Returns the count of compliant and noncompliant conformance packs.
Returns the resource counts across accounts and regions that are compliant.
Returns configuration item that is aggregated for your specific resource.
Returns the evaluation results for the specified Config rule.
Returns the evaluation results for the specified Amazon Web Services resources.
Returns the number of resources that are compliant and noncompliant.
Returns compliance details of a conformance pack for all Amazon Web Services resources.
Returns compliance details for the conformance pack based on the resource type.
Returns the policy definition containing the logic for your Config rule.
Returns the resource types, the number of each resource type, and the compliance status.
Returns detailed status for each member account within an organization.
Returns detailed status for each member account within an organization.
Returns the policy definition containing the logic for your organization.
For accurate reporting on the compliance status, you must record the compliance details.
Returns a summary of resource evaluation for the specified resource.
Returns the details of a specific stored query.
Accepts a resource type and returns a list of resource identifiers that meet the criteria.
Returns a list of conformance pack compliance scores.
Accepts a resource type and returns a list of resource identifiers that meet the criteria.
Returns a list of proactive resource evaluations.
Lists the stored queries for a single Amazon Web Services account.
List the tags for Config resource.
put_aggregation_authorization
put_config_rule
put_configuration_aggregator
put_configuration_recorder
put_conformance_pack
put_delivery_channel
put_evaluations
put_external_evaluation
put_organization_config_rule
put_organization_conformance_pack
put_remediation_configurations
put_remediation_exceptions
put_resource_config
put_retention_configuration
put_stored_query
select_aggregate_resource_config
select_resource_config
start_config_rules_evaluation
start_configuration_recorder
start_remediation_execution
start_resource_evaluation
stop_configuration_recorder
tag_resource
untag_resource

Authorizes the aggregator account and region to collect data from the source account and region
Adds or updates an Config rule to evaluate if your Amazon Web Services resources comply with your desired configurations
Creates and updates the configuration aggregator with the selected source accounts and regions
Creates a new configuration recorder to record configuration changes
Creates or updates a conformance pack
Creates a delivery channel object to deliver configuration information
Used by an Lambda function to deliver evaluation results to Config
Add or updates the evaluations for process checks
Adds or updates a Config rule for your entire organization to evaluate your Amazon Web Services resources
Deploys conformance packs across member accounts in an Amazon Web Services Organization
Adds or updates the remediation configuration with a specific Config rule
A remediation exception is when a specified resource is no longer considered for auto-remediation
Records the configuration state for the resource provided in the request
Saves a new query or updates an existing saved query
Accepts a structured query language (SQL) SELECT command and an aggregator
Accepts a structured query language (SQL) SELECT command, an aggregator
Runs an on-demand evaluation for the specified Config rules against the last known configuration state of the resources
Starts recording configurations of the Amazon Web Services resources
Runs an on-demand remediation for the specified Config rules against the last known remediation configuration
Runs an on-demand evaluation for the specified resource to determine whether its configuration will comply
Stops recording configurations of the Amazon Web Services resources
Associates the specified tags to a resource with the specified resourceArn
Deletes specified tags from a resource

Examples

```r
## Not run:
svc <- configservice()
svc$batch_get_aggregate_resource_config(
    Foo = 123
)

## End(Not run)
```

connect

<table>
<thead>
<tr>
<th>Amazon Connect Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amazon Connect Service</td>
</tr>
</tbody>
</table>

Description

Amazon Connect is a cloud-based contact center solution that you use to set up and manage a customer contact center and provide reliable customer engagement at any scale.

Amazon Connect provides metrics and real-time reporting that enable you to optimize contact routing. You can also resolve customer issues more efficiently by getting customers in touch with the appropriate agents.
There are limits to the number of Amazon Connect resources that you can create. There are also limits to the number of requests that you can make per second. For more information, see Amazon Connect Service Quotas in the Amazon Connect Administrator Guide.

You can connect programmatically to an Amazon Web Services service by using an endpoint. For a list of Amazon Connect endpoints, see Amazon Connect Endpoints.

Usage

connect(config = list(), credentials = list(), endpoint = NULL, region = NULL)

Arguments

config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

directory Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```r
svc <- connect(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- **activate_evaluation_form**: Activates an evaluation form in the specified Amazon Connect instance
  - This API is in preview release for Amazon Connect and is subject to change
- **associate_analytics_data_set**: This API is in preview release for Amazon Connect and is subject to change
- **associate_approved_origin**: This API is in preview release for Amazon Connect and is subject to change
- **associate_bot**: Associates an existing vocabulary as the default
  - Associates a connect resource to a flow
- **associate_default_vocabulary**: This API is in preview release for Amazon Connect and is subject to change
- **associate_flow**: Associates a flow with a phone number claimed to your Amazon Connect instance
  - Associates a set of queues with a routing profile
- **associate_instance_storage_config**: This API is in preview release for Amazon Connect and is subject to change
- **associate_lambda_function**: This API is in preview release for Amazon Connect and is subject to change
- **associate_lex_bot**: Associates an agent with a traffic distribution group
- **associate_phone_number_contact_flow**: This API is in preview release for Amazon Connect and is subject to change
- **associate_queue_quick.connects**: This API is in preview release for Amazon Connect and is subject to change
- **associate_routing_profile_queues**: This API is in preview release for Amazon Connect and is subject to change
- **associate_security_key**: This API is in preview release for Amazon Connect and is subject to change
- **associate_traffic_distribution_group_user**: This API is in preview release for Amazon Connect and is subject to change
connect

associate_user_proficiencies
batch_associate_analytics_data_set
batch_disassociate_analytics_data_set
batch_get_flow_association
batch_put_contact
claim_phone_number
create_agent_status
create_contact_flow
create_contact_flow_module
create_evaluation_form
create_hours_of_operation
create_instance
create_integration_association
create_participant
create_persistent_contact_association
create_predefined_attribute
create_prompt
create_queue
create_quick_connect
create_routing_profile
create_rule
create_security_profile
create_task_template
create_traffic_distribution_group
create_use_case
create_user
create_user_hierarchy_group
create_view
create_view_version
create_vocabulary
deactivate_evaluation_form
delete_contact_evaluation
delete_contact_flow
delete_contact_flow_module
delete_evaluation_form
delete_hours_of_operation
delete_instance
delete_integration_association
delete_predefined_attribute
delete_prompt
delete_queue
delete_quick_connect
delete_routing_profile
delete_rule
delete_security_profile
delete_task_template
delete_traffic_distribution_group
delete_use_case

>Associates a set of proficiencies with a user
This API is in preview release for Amazon Connect and is subject to change
Retrieves the flow associations for the given resources
Only the Amazon Connect outbound campaigns service principal is allowed
Claims an available phone number to your Amazon Connect instance or traffic distribution group
This API is in preview release for Amazon Connect and is subject to change
Creates a flow for the specified Amazon Connect instance
Creates a flow module for the specified Amazon Connect instance
Creates an evaluation form in the specified Amazon Connect instance
This API is in preview release for Amazon Connect and is subject to change
This API is in preview release for Amazon Connect and is subject to change
Creates an Amazon Web Services resource association with an Amazon Connect instance
Adds a new participant into an on-going chat contact
Enables rehydration of chats for the lifespan of a contact
Creates a new predefined attribute for the specified Amazon Connect instance
Creates a prompt
This API is in preview release for Amazon Connect and is subject to change
Creates a quick connect for the specified Amazon Connect instance
Creates a new routing profile
Creates a rule for the specified Amazon Connect instance
Creates a security profile
Creates a new task template in the specified Amazon Connect instance
Creates a traffic distribution group given an Amazon Connect instance that has been replicated
Creates a use case for an integration association
Creates a user account for the specified Amazon Connect instance
Creates a new user hierarchy group
Creates a new view with the possible status of SAVED or PUBLISHED
Publishes a new version of the view identifier
Creates a custom vocabulary associated with your Amazon Connect instance
Deactivates an evaluation form in the specified Amazon Connect instance
Deletes a contact evaluation in the specified Amazon Connect instance
Deletes a flow for the specified Amazon Connect instance
Deletes the specified flow module
Deletes an evaluation form in the specified Amazon Connect instance
This API is in preview release for Amazon Connect and is subject to change
This API is in preview release for Amazon Connect and is subject to change
Deletes an Amazon Web Services resource association from an Amazon Connect instance
Deletes a predefined attribute from the specified Amazon Connect instance
Deletes a prompt
Deletes a queue
Deletes a quick connect
Deletes a routing profile
Deletes a rule for the specified Amazon Connect instance
Deletes a security profile
Deletes the task template
Deletes a traffic distribution group
Deletes a use case from an integration association
delete_user Deletes a user account from the specified Amazon Connect instance
delete_user_hierarchy_group Deletes an existing user hierarchy group
delete_view Deletes the view entirely
delete_view_version Deletes the particular version specified in ViewVersion identifier
delete_vocabulary Deletes the vocabulary that has the given identifier
describe_agent_status This API is in preview release for Amazon Connect and is subject to change
describe_contact This API is in preview release for Amazon Connect and is subject to change
describe_contact_evaluation Describes a contact evaluation in the specified Amazon Connect instance
describe_contact_flow Describes the specified flow
describe_contact_flow_module Describes the specified flow module
describe_evaluation_form Describes an evaluation form in the specified Amazon Connect instance
describe_instance This API is in preview release for Amazon Connect and is subject to change
describe_instance_attribute Describes a predefined attribute for the specified Amazon Connect instance
describe_instance_storage_config Describes the prompt
describe_queue Describes the queue
describe_quick_connect Describes the quick connect
describe_routing_profile Describes the specified routing profile
describe_rule Describes a rule for the specified Amazon Connect instance
describe_security_profile Gets basic information about the security profile
describe_traffic_distribution_group Gets details and status of a traffic distribution group
describe_user Describes the specified user
describe_user_hierarchy_group Describes the specified hierarchy group
describe_user_hierarchy_structure Describes the hierarchy structure of the specified Amazon Connect instance
describe_view Retrieves the view for the specified Amazon Connect instance and view identifier
describe_vocabulary Describes the specified vocabulary
disassociate_analytics_data_set This API is in preview release for Amazon Connect and is subject to change
disassociate_approved_origin This API is in preview release for Amazon Connect and is subject to change
disassociate_bot This API is in preview release for Amazon Connect and is subject to change
disassociate_flow Disassociates a connect resource from a flow
disassociate_instance_storage_config This API is in preview release for Amazon Connect and is subject to change
disassociate_lambda_function This API is in preview release for Amazon Connect and is subject to change
disassociate_lex_bot This API is in preview release for Amazon Connect and is subject to change
disassociate_phone_number_contact_flow Removes the flow association from a phone number claimed to your Amazon Connect instance
disassociate_queue_quick_connects This API is in preview release for Amazon Connect and is subject to change
disassociate_routing_profile_queues This API is in preview release for Amazon Connect and is subject to change
disassociate_security_key This API is in preview release for Amazon Connect and is subject to change
disassociate_traffic_distribution_group_user This API is in preview release for Amazon Connect and is subject to change
disassociate_user_proficiencies This API is in preview release for Amazon Connect and is subject to change
dismiss_user_contact Removes contacts from an agent’s CCP and returns the agent to an available state
get_contact_attributes Retrieves current contact attributes
get_current_metric_data Gets the real-time metric data from the specified Amazon Connect instance
get_current_user_data Gets the real-time active user data from the specified Amazon Connect instance
get_federation_token Supports SAML sign-in for Amazon Connect
get_flow_association Retrieves the flow associated for a given resource
connect

get_metric_data
get_metric_data_v2
get_prompt_file
get_task_template
get_traffic_distribution
import_phone_number
list_agent_statuses
list_analytics_data_associations
list_approved_origins
list_bots
list_contact_evaluations
list_contact_flow_modules
list_contact_flows
list_contact_references
list_default_vocabularies
list_evaluation_forms
list_evaluation_form_versions
list_flow_associations
list_hours_of_operations
list_instance_attributes
list_instances
list_instance_storage_configs
list_integration_associations
list_lambda_functions
list_lex_bots
list_phone_numbers
list_phone_numbers_v2
list_predefined_attributes
list_prompts
list_queue_quick_connects
list_queues
list_quick_connects
list_realtime_contact_analysis_segments_v2
list_routing_profile_queues
list_routing_profiles
list_rules
list_security_keys
list_security_profile_applications
list_security_profile_permissions
list_security_profiles
list_tags_for_resource
list_task_templates
list_traffic_distribution_groups
list_traffic_distribution_group_users
list_use_cases
list_user_hierarchy_groups
list_user_proficiencies
list_users

 Gets historical metric data from the specified Amazon Connect instance
 Gets metric data from the specified Amazon Connect instance
 Gets the prompt file
 Gets details about a specific task template in the specified Amazon Connect instance
 Retrieves the current traffic distribution for a given traffic distribution group
 Imports a claimed phone number from an external service, such as Amazon Pinpoint
 This API is in preview release for Amazon Connect and is subject to change
 This API is in preview release for Amazon Connect and is subject to change
 This API is in preview release for Amazon Connect and is subject to change
 This API is in preview release for Amazon Connect and is subject to change
 Lists contact evaluations in the specified Amazon Connect instance
 Provides information about the flow modules for the specified Amazon Connect instance
 Provides information about the flows for the specified Amazon Connect instance
 This API is in preview release for Amazon Connect and is subject to change
 Lists the default vocabularies for the specified Amazon Connect instance
 Lists evaluation forms in the specified Amazon Connect instance
 Lists versions of an evaluation form in the specified Amazon Connect instance
 List the flow association based on the filters
 Provides information about the hours of operation for the specified Amazon Connect instance
 This API is in preview release for Amazon Connect and is subject to change
 This API is in preview release for Amazon Connect and is subject to change
 This API is in preview release for Amazon Connect and is subject to change
 Provides summary information about the Amazon Web Services resource associations for the specified Amazon Connect instance
 This API is in preview release for Amazon Connect and is subject to change
 Provides information about the phone numbers for the specified Amazon Connect instance
 Lists predefined attributes for the specified Amazon Connect instance
 Provides information about the prompts for the specified Amazon Connect instance
 Provides information about the queues for the specified Amazon Connect instance
 Provides information about the quick connects for the specified Amazon Connect instance
 Provides a list of analysis segments for a real-time analysis session
 Lists the queues associated with a routing profile
 Provides summary information about the routing profiles for the specified Amazon Connect instance
 List all rules for the specified Amazon Connect instance
 This API is in preview release for Amazon Connect and is subject to change
 Returns a list of third-party applications in a specific security profile
 Lists the permissions granted to a security profile
 Provides summary information about the security profiles for the specified Amazon Connect instance
 Lists the tags for the specified resource
 Lists task templates for the specified Amazon Connect instance
 Lists traffic distribution groups
 Lists traffic distribution group users
 Lists the use cases for the integration association
 Provides summary information about the hierarchy groups for the specified Amazon Connect instance
 Lists proficiencies associated with a user
 Provides summary information about the users for the specified Amazon Connect instance
list_views
list_view_versions
monitor_contact
pause_contact
put_user_status
release_phone_number
replicate_instance
resume_contact
resume_contact_recording
search_available_phone_numbers
search_contacts
search_hours_of_operations
search_predefined_attributes
search_prompts
search_queues
search_quick_connects
search_resource_tags
search_routing_profiles
search_security_profiles
search_users
search_vocabularies
send_chat_integration_event
start_chat_contact
start_contact_evaluation
start_contact_recording
start_contact_streaming
start_outbound_voice_contact
start_task_contact
start_web_rtc_contact
stop_contact
stop_contact_recording
stop_contact_streaming
submit_contact_evaluation
suspend_contact_recording
tag_contact
tag_resource
transfer_contact
untag_contact
untag_resource
update_agent_status
update_contact
update_contact_attributes
update_contact_evaluation
update_contact_flow_content
update_contact_flow_metadata
update_contact_flow_module_content
update_contact_flow_module_metadata
update_contact_flow_name

Returns views in the given instance
Returns all the available versions for the specified Amazon Connect instance
Initiates silent monitoring of a contact
Allows pausing an ongoing task contact
Changes the current status of a user or agent in Amazon Connect
Releases a phone number previously claimed to an Amazon Connect instance
Replicates an Amazon Connect instance in the specified Amazon Web Services Region
Allows resuming a task contact in a paused state
When a contact is being recorded, and the recording has been suspended using SuspendContactRecording, this API resumes recording whatever recording is selected in the flow configuration: call, screen, or both
Searches for available phone numbers that you can claim to your Amazon Connect instance
Searches contacts in an Amazon Connect instance
Searches the hours of operation in an Amazon Connect instance, with optional filtering
Searches prompts in an Amazon Connect instance, with optional filtering
Searches queues in an Amazon Connect instance
Searches quick connects in an Amazon Connect instance, with optional filtering
Searches tags used in an Amazon Connect instance using optional search criteria
Searches routing profiles in an Amazon Connect instance, with optional filtering
Searches security profiles in an Amazon Connect instance, with optional filtering
Searches users in an Amazon Connect instance, with optional filtering
Searches for vocabularies within a specific Amazon Connect instance using State, NameStartsWith, and LanguageCode
Processes chat integration events from Amazon Web Services or external integrations
Initiates a flow to start a new chat for the customer
Starts an empty evaluation in the specified Amazon Connect instance, using the specified evaluation form
Starts recording the contact:
Initiates real-time message streaming for a new chat contact
Places an outbound call to a contact, and then initiates the flow
Creates or updates user-defined contact attributes associated with the specified contact
Updates metadata about the specified flow
update_contact_routing_data
update_contact_schedule
update_evaluation_form
update_hours_of_operation
update_instance_attribute
update_instance_storage_config
update_participant_role_config
update_phone_number
update_phone_number_metadata
update_predefined_attribute
update_prompt
update_queue_hours_of_operation
update_queue_max_contacts
update_queue_name
update_queue_outbound_caller_config
update_queue_status
update_quick_connect_config
update_quick_connect_name
update_routing_profile_agent_availability_timer
update_routing_profile_concurrency
update_routing_profile_default_outbound_queue
update_routing_profile_name
update_routing_profile_queues
update_rule
update_security_profile
update_task_template
update_traffic_distribution
update_user_hierarchy
update_user_hierarchy_group_name
update_user_hierarchy_structure
update_user_identity_info
update_user_phone_config
update_user_proficiencies
update_user_routing_profile
update_user_security_profiles
update_view_content
update_view_metadata

This API is in preview release for Amazon Connect and is subject to change.
Updates the scheduled time of a task contact that is already scheduled.
Updates details about a specific evaluation form version in the specified Amazon Connect instance.
This API is in preview release for Amazon Connect and is subject to change.
Updates timeouts for when human chat participants are to be considered idle.
Updates your claimed phone number from its current Amazon Connect instance.
Updates a phone number’s metadata.
Updates a predefined attribute for the specified Amazon Connect instance.
Updates a prompt.
This API is in preview release for Amazon Connect and is subject to change.
Updates the channels that agents can handle in the Contact Control Panel (CCP).
Updates the default outbound queue of a routing profile.
Updates the name and description of a routing profile.
Updates the properties associated with a set of queues for a routing profile.
Updates a rule for the specified Amazon Connect instance.
Updates a security profile.
Updates details about a specific task template in the specified Amazon Connect instance.
Updates the traffic distribution for a given traffic distribution group.
Assigns the specified hierarchy group to the specified user.
Updates the name of the user hierarchy group.
Updates the user hierarchy structure: add, remove, and rename user hierarchy levels.
Updates the identity information for the specified user.
Updates the phone configuration settings for the specified user.
Updates the properties associated with the proficiencies of a user.
Assigns the specified routing profile to the specified user.
Assigns the specified security profiles to the specified user.
Updates the view content of the given view identifier in the specified Amazon Connect instance.
Updates the view metadata.

Examples

```r
## Not run:
svc <- connect()
svc$activate_evaluation_form(
  Foo = 123
)

## End(Not run)
```
**connectcampaignservice**

*AmazonConnectCampaignService*

**Description**

Provide APIs to create and manage Amazon Connect Campaigns.

**Usage**

```python
cconnectcampaignservice(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

**Arguments**

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
connectcampaignservice

- **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**
A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- connectcampaignservice(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

**Operations**
create_campaign
delete_campaign
delete_connect_instance_config
delete_instance_onboarding_job
describe_campaign
get_campaign_state
get_campaign_state_batch
get_connect_instance_config
get_instance_onboarding_job_status
list_campaigns
list_tags_for_resource
pause_campaign
put_dial_request_batch
resume_campaign
start_campaign
start_instance_onboarding_job
stop_campaign
tag_resource
untag_resource
update_campaign_dialer_config
update_campaign_name
update_campaign_outbound_call_config

Examples

```r
## Not run:
svc <- connectcampaignservice()
svc$create_campaign(
  Foo = 123
)

## End(Not run)
```

connectcases

### Amazon Connect Cases

#### Description

With Amazon Connect Cases, your agents can track and manage customer issues that require multiple interactions, follow-up tasks, and teams in your contact center. A case represents a customer issue. It records the issue, the steps and interactions taken to resolve the issue, and the outcome. For more information, see Amazon Connect Cases in the Amazon Connect Administrator Guide.
connectcases

Usage

connectcases(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - credentials:
    - creds:
      - access_key_id: AWS access key ID
      - secret_access_key: AWS secret access key
      - session_token: AWS temporary session token
    - profile: The name of a profile to use. If not given, then the default profile is used.
    - anonymous: Set anonymous credentials.
  - endpoint: The complete URL to use for the constructed client.
  - region: The AWS Region used in instantiating the client.
  - close_connection: Immediately close all HTTP connections.
  - timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

- **credentials**: Optional credentials shorthand for the config parameter
  - creds:
    - access_key_id: AWS access key ID
    - secret_access_key: AWS secret access key
    - session_token: AWS temporary session token
    - profile: The name of a profile to use. If not given, then the default profile is used.
    - anonymous: Set anonymous credentials.

- **endpoint**: Optional shorthand for complete URL to use for the constructed client.

- **region**: Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```r
svc <- connectcases(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- **batch_get_field**: Returns the description for the list of fields in the request parameters
- **batch_put_field_options**: Creates and updates a set of field options for a single select field in a Cases domain
- **create_case**: If you provide a value for PerformedBy
- **create_domain**: Creates a domain, which is a container for all case data, such as cases, fields, templates and layouts
- **create_field**: Creates a field in the Cases domain
- **create_layout**: Creates a layout in the Cases domain
- **create_related_item**: Creates a related item (comments, tasks, and contacts) and associates it with a case
- **create_template**: Creates a template in the Cases domain
- **delete_domain**: Deletes a Cases domain
- **delete_field**: Deletes a field from a cases template
- **delete_layout**: Deletes a layout from a cases template
- **delete_template**: Deletes a cases template
- **get_case**: Returns information about a specific case if it exists
- **get_case_audit_events**: Returns the audit history about a specific case if it exists
get_case_event_configuration | Returns the case event publishing configuration
get_domain | Returns information about a specific domain if it exists
get_layout | Returns the details for the requested layout
get_template | Returns the details for the requested template
list_cases_for_contact | Lists cases for a given contact
list_domains | Lists all cases domains in the Amazon Web Services account
list_field_options | Lists all of the field options for a field identifier in the domain
list_fields | Lists all fields in a Cases domain
list_layouts | Lists all layouts in the given cases domain
list_tags_for_resource | Lists tags for a resource
listTemplates | Lists all of the templates in a Cases domain
put_case_event_configuration | Adds case event publishing configuration
search_cases | Searches for cases within their associated Cases domain
search_related_items | Searches for related items that are associated with a case
tag_resource | Adds tags to a resource
untag_resource | Untags a resource
update_case | If you provide a value for PerformedBy
update_field | Updates the properties of an existing field
update_layout | Updates the attributes of an existing layout
update_template | Updates the attributes of an existing template

Examples

```r
## Not run:
svc <- connectcases()
svc$batch_get_field(
  Foo = 123
)

## End(Not run)
```

**Description**

Contact Lens for Amazon Connect enables you to analyze conversations between customer and agents, by using speech transcription, natural language processing, and intelligent search capabilities. It performs sentiment analysis, detects issues, and enables you to automatically categorize contacts.

Contact Lens for Amazon Connect provides both real-time and post-call analytics of customer-agent conversations. For more information, see [Analyze conversations using Contact Lens](#) in the *Amazon Connect Administrator Guide*. 
connectcontactlens

Usage

```r
connectcontactlens(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **creds**:
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
    - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.
- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
    - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

- **endpoint**: Optional shorthand for complete URL to use for the constructed client.
- **region**: Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```r
svc <- connectcontactlens(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- **list_realtime_contact_analysis_segments**  Provides a list of analysis segments for a real-time analysis session

Examples

```r
## Not run:
svc <- connectcontactlens()
svc$list_realtime_contact_analysis_segments(
  Foo = 123
)
```

## End(Not run)
Amazon Connect Participant Service

Description

Amazon Connect is an easy-to-use omnichannel cloud contact center service that enables companies of any size to deliver superior customer service at a lower cost. Amazon Connect communications capabilities make it easy for companies to deliver personalized interactions across communication channels, including chat.

Use the Amazon Connect Participant Service to manage participants (for example, agents, customers, and managers listening in), and to send messages and events within a chat contact. The APIs in the service enable the following: sending chat messages, attachment sharing, managing a participant’s connection state and message events, and retrieving chat transcripts.

Usage

```r
connectparticipant(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)
connectparticipant

credentials Optional credentials shorthand for the config parameter

  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

default Optional credentials shorthand for the config parameter

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- connectparticipant(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
)```
region = "string"

Operations

- **complete_attachment_upload**
  Allows you to confirm that the attachment has been uploaded using the pre-signed URL provided in StartAttachmentUpload API

- **create_participant_connection**
  Creates the participant’s connection

- **describe_view**
  Retrieves the view for the specified view token

- **disconnect_participant**
  Disconnects a participant

- **get_attachment**
  Provides a pre-signed URL for download of a completed attachment

- **get_transcript**
  Retrieves a transcript of the session, including details about any attachments

- **send_event**
  The application/vnd

- **send_message**
  Sends a message

- **start_attachment_upload**
  Provides a pre-signed Amazon S3 URL in response for uploading the file directly to S3

Examples

```r
## Not run:
svc <- connectparticipant()
svc$complete_attachment_upload(
  Foo = 123
)

## End(Not run)
```

---

**connectwisdomservice**  
Amazon Connect Wisdom Service

Description

Amazon Connect Wisdom delivers agents the information they need to solve customer issues as they’re actively speaking with customers. Agents can search across connected repositories from within their agent desktop to find answers quickly. Use Amazon Connect Wisdom to create an assistant and a knowledge base, for example, or manage content by uploading custom files.

Usage

```r
connectwisdomservice(  
  config = list(),  
  credentials = list(),  
  endpoint = NULL,  
  region = NULL  
)
```
Arguments

config Optional configuration of credentials, endpoint, and/or region.

- credentials:
  - creds:
    * access_key_id: AWS access key ID
    * secret_access_key: AWS secret access key
    * session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.

- endpoint: The complete URL to use for the constructed client.
- region: The AWS Region used in instantiating the client.
- close_connection: Immediately close all HTTP connections.
- timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/Bucket/KEY.
- sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter

- creds:
  - access_key_id: AWS access key ID
  - secret_access_key: AWS secret access key
  - session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```
svc <- connectwisdomservice(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
      )
    )
  )
)```
secret_access_key = "string",
session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string",
close_connection = "logical",
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
creds = list(
access_key_id = "string",
secret_access_key = "string",
session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

create_assistant
create_assistant_association
create_content
create_knowledge_base
create_quick_response
create_session
delete_assistant
delete_assistant_association
delete_content
delete_import_job
delete_knowledge_base
delete_quick_response
get_assistant
get_assistant_association
get_content
get_content_summary
get_import_job
get_knowledge_base
get_quick_response
get_recommendations

Creates an Amazon Connect Wisdom assistant
Creates an association between an Amazon Connect Wisdom assistant and another resource
Creates Wisdom content
Creates a knowledge base
Creates a Wisdom quick response
Creates a session
Deletes an assistant
Deletes an assistant association
Deletes the content
Deletes the quick response import job
Deletes the knowledge base
Deletes a quick response
Retrieves information about an assistant
Retrieves information about an assistant association
Retrieves content, including a pre-signed URL to download the content
Retrieves summary information about the content
Retrieves the started import job
Retrieves information about the knowledge base
Retrieves the quick response
Retrieves recommendations for the specified session
get_session Retrieves information for a specified session
list_assistant_associations Lists information about assistant associations
list_assistants Lists information about assistants
list_contents Lists the content
list_import_jobs Lists information about import jobs
list_knowledge_bases Lists the knowledge bases
list_quick_responses Lists information about quick response
list_tags_for_resource Lists the tags for the specified resource
notify_recommendations_received Removes the specified recommendations from the specified assistant’s queue of new
query_assistant Performs a manual search against the specified assistant
remove_knowledge_base_template_uri Removes a URI template from a knowledge base
search_content Searches for content in a specified knowledge base
search_quick_responses Searches existing Wisdom quick responses in a Wisdom knowledge base
search_sessions Searches for sessions
start_content_upload Get a URL to upload content to a knowledge base
start_import_job Start an asynchronous job to import Wisdom resources from an uploaded source file
tag_resource Adds the specified tags to the specified resource
untag_resource Removes the specified tags from the specified resource
update_content Updates information about the content
update_knowledge_base_template_uri Updates the template URI of a knowledge base
update_quick_response Updates an existing Wisdom quick response

Examples

```r
## Not run:
svc <- connectwisdomservice()
svc$create_assistant(
  Foo = 123
)
## End(Not run)
```

Description

These interfaces allow you to apply the Amazon Web Services library of pre-defined controls to your organizational units, programmatically. In Amazon Web Services Control Tower, the terms "control" and "guardrail" are synonyms.

To call these APIs, you’ll need to know:

- the controlIdentifier for the control—or guardrail—you are targeting.
- the ARN associated with the target organizational unit (OU), which we call the targetIdentifier.
• the ARN associated with a resource that you wish to tag or untag.

**To get the controlIdentifier for your Amazon Web Services Control Tower control:**

The controlIdentifier is an ARN that is specified for each control. You can view the controlIdentifier in the console on the **Control details** page, as well as in the documentation.

The controlIdentifier is unique in each Amazon Web Services Region for each control. You can find the controlIdentifier for each Region and control in the **Tables of control metadata** in the Amazon Web Services Control Tower User Guide.

A quick-reference list of control identifiers for the Amazon Web Services Control Tower legacy **Strongly recommended** and **Elective** controls is given in Resource identifiers for APIs and controls in the Controls reference guide section of the Amazon Web Services Control Tower User Guide. Remember that **Mandatory** controls cannot be added or removed.

**ARN format:** arn:aws:controltower:{REGION}::control/{CONTROL_NAME}

**Example:**

arn:aws:controltower:us-west-2::control/AWS-GR_AUTOSCALING_LAUNCH_CONFIG_PUBLIC_IP_DISABLED

**To get the targetIdentifier:**

The targetIdentifier is the ARN for an OU.

In the Amazon Web Services Organizations console, you can find the ARN for the OU on the Organizational unit details page associated with that OU.

**OU ARN format:**

arn:{Partition}:organizations::${MasterAccountId}:ou/o-$OrganizationId/ou-$OrganizationalUnitId

**Details and examples**

• Control API input and output examples with CLI

• Enable controls with CloudFormation

• Control metadata tables

• List of identifiers for legacy controls

• Controls reference guide

• Controls library groupings

• Creating Amazon Web Services Control Tower resources with Amazon Web Services CloudFormation

To view the open source resource repository on GitHub, see aws-cloudformation/aws-cloudformation-resource-providers-controltower

**Recording API Requests**

Amazon Web Services Control Tower supports Amazon Web Services CloudTrail, a service that records Amazon Web Services API calls for your Amazon Web Services account and delivers log files to an Amazon S3 bucket. By using information collected by CloudTrail, you can determine which requests the Amazon Web Services Control Tower service received, who made the request and when, and so on. For more about Amazon Web Services Control Tower and its support for CloudTrail, see Logging Amazon Web Services Control Tower Actions with Amazon Web Services CloudTrail in the Amazon Web Services Control Tower User Guide. To learn more about CloudTrail, including how to turn it on and find your log files, see the Amazon Web Services CloudTrail User Guide.
usage

```r
controltower(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **creds**:
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
    - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
    - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

- **endpoint**: Optional shorthand for complete URL to use for the constructed client.
- **region**: Optional shorthand for AWS Region used in instantiating the client.

value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```r
svc <- controltower(
    config = list(
        credentials = list(
            creds = list(
                access_key_id = "string",
                secret_access_key = "string",
                session_token = "string"
            ),
            profile = "string",
            anonymous = "logical"
        ),
        endpoint = "string",
        region = "string",
        close_connection = "logical",
        timeout = "numeric",
        s3_force_path_style = "logical",
        sts_regional_endpoint = "string"
    ),
    credentials = list(
        creds = list(
            access_key_id = "string",
            secret_access_key = "string",
            session_token = "string"
        ),
        profile = "string",
        anonymous = "logical"
    ),
    endpoint = "string",
    region = "string"
)
```

Operations

- `create_landing_zone`: Creates a new landing zone
- `delete_landing_zone`: Decommissions a landing zone
- `disable_baseline`: Disable an EnabledBaseline resource on the specified Target
- `disable_control`: This API call turns off a control
- `enable_baseline`: Enable (apply) a Baseline to a Target
- `enable_control`: This API call activates a control
- `get_baseline`: Retrieve details about an existing Baseline resource by specifying its identifier
- `get_baseline_operation`: Returns the details of an asynchronous baseline operation, as initiated by any of these APIs: EnableBaseline, DisableBaseline, UpdateEnabledBaseline, ResetEnabledBaseline
- `get_control_operation`: Returns the status of a particular EnableControl or DisableControl operation
- `get_enabled_baseline`: Retrieve details of an EnabledBaseline resource by specifying its identifier
- `get_enabled_control`: Retrieves details about an enabled control
- `get_landing_zone`: Returns details about the landing zone
- `get_landing_zone_operation`: Returns the status of the specified landing zone operation
- `list_baselines`: Returns a summary list of all available baselines
## Not run:

```r
svc <- controltower()
svc$create_landing_zone(
  Foo = 123
)
```

## End(Not run)

---

**costandusagereportservice**

AWS Cost and Usage Report Service

**Description**

You can use the Amazon Web Services Cost and Usage Report API to programmatically create, query, and delete Amazon Web Services Cost and Usage Report definitions.

Amazon Web Services Cost and Usage Report track the monthly Amazon Web Services costs and usage associated with your Amazon Web Services account. The report contains line items for each unique combination of Amazon Web Services product, usage type, and operation that your Amazon Web Services account uses. You can configure the Amazon Web Services Cost and Usage Report to show only the data that you want, using the Amazon Web Services Cost and Usage Report API.

**Service Endpoint**

The Amazon Web Services Cost and Usage Report API provides the following endpoint:

- cur.us-east-1.amazonaws.com
Usage

costandusagereportservice(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

costandusagereportservice

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **creds**:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.
- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - **anonymous**: Set anonymous credentials.

- **endpoint**: Optional shorthand for complete URL to use for the constructed client.
- **region**: Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```r
csvc <- costandusagereportservice(
    config = list(
        credentials = list(
            creds = list(
                access_key_id = "string",
                secret_access_key = "string",
                session_token = "string"
            ),
            profile = "string",
            anonymous = "logical"
        ),
        endpoint = "string",
        region = "string",
        close_connection = "logical",
        timeout = "numeric",
        s3_force_path_style = "logical",
        sts_regional_endpoint = "string"
    ),
    credentials = list(
        creds = list(
            access_key_id = "string",
            secret_access_key = "string",
            session_token = "string"
        ),
        profile = "string",
        anonymous = "logical"
    ),
    endpoint = "string",
    region = "string"
)
```

Operations

- `delete_report_definition`: Deletes the specified report
- `describe_report_definitions`: Lists the Amazon Web Services Cost and Usage Report available to this account
- `list_tags_for_resource`: Lists the tags associated with the specified report definition
- `modify_report_definition`: Allows you to programmatically update your report preferences
- `put_report_definition`: Creates a new report using the description that you provide
- `tag_resource`: Associates a set of tags with a report definition
- `untag_resource`: Disassociates a set of tags from a report definition

Examples

```r
## Not run:
svc <- costandusagereportservice()
```
# The following example deletes the AWS Cost and Usage report named
# ExampleReport.
svc$delete_report_definition(
    ReportName = "ExampleReport"
)

## End(Not run)

costexplorer  
AWS Cost Explorer Service

Description

You can use the Cost Explorer API to programmatically query your cost and usage data. You can query for aggregated data such as total monthly costs or total daily usage. You can also query for granular data. This might include the number of daily write operations for Amazon DynamoDB database tables in your production environment.

Service Endpoint

The Cost Explorer API provides the following endpoint:

  • https://ce.us-east-1.amazonaws.com

For information about the costs that are associated with the Cost Explorer API, see Amazon Web Services Cost Management Pricing.

Usage

costexplorer(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

config  
Optional configuration of credentials, endpoint, and/or region.

• credentials:
  – creds:
    • access_key_id: AWS access key ID
    • secret_access_key: AWS secret access key
    • session_token: AWS temporary session token
  – profile: The name of a profile to use. If not given, then the default profile is used.
  – anonymous: Set anonymous credentials.
• endpoint: The complete URL to use for the constructed client.
• **region**: The AWS Region used in instantiating the client.
• **close_connection**: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ende.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ende.html)

**credentials** Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- costexplorer(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
```

• region: The AWS Region used in instantiating the client.
• close_connection: Immediately close all HTTP connections.
• timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ende.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ende.html)

**credentials** Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- costexplorer(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
```
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

create_anomaly_monitor
create_anomaly_subscription
create_cost_category_definition
delete_anomaly_monitor
delete_anomaly_subscription
delete_cost_category_definition
describe_cost_category_definition
get_anomalies
get_anomaly_monitors
get_anomaly_subscriptions
get_approximate_usage_records
get_cost_and_usage
get_cost_and_usage_with_resources
get_cost_categories
get_cost_forecast
get_dimension_values
get_reservation_coverage
get_reservation_purchase_recommendation
get_reservation_utilization
get_rightsizing_recommendation
get_savings_plan_purchase_recommendation_details
get_savings_plans_coverage
get_savings_plans_purchase_recommendation
get_savings_plans_utilization
get_savings_plans_utilization_details
get_tags
get_usage_forecast
list_cost_allocation_tag_backfill_history
list_cost_allocation_tags
list_cost_category_definitions
list_savings_plan_purchase_recommendation_generation
list_tags_for_resource
provide_anomaly_feedback

Creates a new cost anomaly detection monitor with the requested type and monitor specification
Adds an alert subscription to a cost anomaly detection monitor
Creates a new Cost Category with the requested name and rules
Deletes a cost anomaly monitor
Deletes a cost anomaly subscription
Deletes a Cost Category
Retrieves the name, Amazon Resource Name (ARN), rules, definition, and effective dates of a Cost Category
Retrieves all of the cost anomalies detected on your account during the time period that's specified by the DateInterval object
Retrieves the cost anomaly monitor definitions for your account
Retrieves the cost anomaly subscription objects for your account
Retrieves estimated usage records for hourly granularity or resource-level data at daily granularity
Retrieves cost and usage metrics for your account
Retrieves cost and usage metrics with resources for your account
Retrieves an array of Cost Category names and values incurred during a specified period
Retrieves a forecast for how much Amazon Web Services predicts that you will spend over the forecast time period that you select, based on your past costs
Retrieves the reservation coverage for your account, which you can use to see how much of your Amazon Elastic Compute Cloud, Amazon ElastiCache, Amazon Relational Database Service, or Amazon Redshift usage is covered by a reservation
Gets recommendations for reservation purchases
Retrieves the reservation utilization for your account
Creates recommendations that help you save cost by identifying idle and underutilized Amazon EC2 instances
Retrieves the details for a Savings Plan recommendation
Retrieves the Savings Plans covered for your account
Retrieves the Savings Plans recommendations for your account
Retrieves the Savings Plans utilization for your account across different periods
Retrieves attribute data along with aggregate utilization and savings from your AWS resources
Retrieves a forecast for how much Amazon Web Services predicts that you will use over the forecast time period that you select, based on your past usage
Retrieves a list of your historical cost allocation tag backfill requests
Retrieves a list of resource tags associated with the resource specified by the Amazon Resource Name (ARN), NumberOfOffices
Modifies the feedback property of a given cost anomaly
customerprofiles

## Not run:
```r
svc <- costexplorer()
svc$create_anomaly_monitor(
  Foo = 123
)

## End(Not run)
```

The `costexplorer()` function can be used to interact with AWS services, allowing for various operations such as creating anomaly monitors.

### Examples

```
## Not run:
svc <- costexplorer()
svc$create_anomaly_monitor(
  Foo = 123
)

## End(Not run)
```

---

**customerprofiles**

**Amazon Connect Customer Profiles**

**Description**

Amazon Connect Customer Profiles is a unified customer profile for your contact center that has pre-built connectors powered by AppFlow that make it easy to combine customer information from third party applications, such as Salesforce (CRM), ServiceNow (ITSM), and your enterprise resource planning (ERP), with contact history from your Amazon Connect contact center. If you’re new to Amazon Connect, you might find it helpful to review the Amazon Connect Administrator Guide.

**Usage**

```r
customerprofiles(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

**Arguments**

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:

---

---

```
customerprofiles(  
  config = list(),  
  credentials = list(),  
  endpoint = NULL,  
  region = NULL  
)
```

---

**customerprofiles**

**Amazon Connect Customer Profiles**

**Description**

Amazon Connect Customer Profiles is a unified customer profile for your contact center that has pre-built connectors powered by AppFlow that make it easy to combine customer information from third party applications, such as Salesforce (CRM), ServiceNow (ITSM), and your enterprise resource planning (ERP), with contact history from your Amazon Connect contact center. If you’re new to Amazon Connect, you might find it helpful to review the Amazon Connect Administrator Guide.

**Usage**

```r
customerprofiles(  
  config = list(),  
  credentials = list(),  
  endpoint = NULL,  
  region = NULL  
)
```

**Arguments**

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:

---

---

```
customerprofiles(  
  config = list(),  
  credentials = list(),  
  endpoint = NULL,  
  region = NULL  
)
```
customerprofiles

* access_key_id: AWS access key ID
* secret_access_key: AWS secret access key
* session_token: AWS temporary session token

– profile: The name of a profile to use. If not given, then the default profile is used.

– anonymous: Set anonymous credentials.

• endpoint: The complete URL to use for the constructed client.
• region: The AWS Region used in instantiating the client.
• close_connection: Immediately close all HTTP connections.
• timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter

• creds:
  – access_key_id: AWS access key ID
  – secret_access_key: AWS secret access key
  – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.

  • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

svc <- customerprofiles(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
customerprofiles

endpoint = "string",
region = "string",
close_connection = "logical",
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
)
,
credentials = list(
  creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
  ),
  profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

add_profile_key Associates a new key value with a specific profile, such as a Contact Record ContactId
create_calculated_attribute_definition Creates a new calculated attribute definition
create_domain Creates a domain, which is a container for all customer data, such as customer profile attributes, object types, profile keys, and encryption keys
create_event_stream Creates an event stream, which is a subscription to real-time events, such as when profiles are created and updated through Amazon Connect Customer Profiles
create_integration_workflow Creates an integration workflow
create_profile Creates a standard profile
delete_calculated_attribute_definition Deletes an existing calculated attribute definition
delete_domain Deletes a specific domain and all of its customer data, such as customer profile attributes, object types, profile keys, and encryption keys
delete_event_stream Disables and deletes the specified event stream
delete_integration Removes an integration from a domain
delete_profile Removes the standard customer profile and all data pertaining to the profile
delete_profile_key Removes a searchable key from a customer profile
delete_profile_object Removes an object associated with a profile of a given ProfileObjectType
delete_profile_object_type Removes a ProfileObjectType from a specific domain as well as removes all the ProfileObjects of that type
delete_workflow Deletes the specified workflow and all its corresponding resources
detect_profile_object_type The process of detecting profile object type mapping by using given objects
detect_profile_object_type The process of detecting profile object type mapping by using given objects
get_auto_merging_preview Tests the auto-merging settings of your Identity Resolution Job without merging your data
get_calculated_attribute_definition Provides more information on a calculated attribute definition for Customer Profiles
get_calculated_attribute_for_profile Retrieve a calculated attribute for a customer profile
get_domain Returns information about a specific domain
get_event_stream Returns information about the specified event stream in a specific domain
get_identity_resolution_job Returns information about an Identity Resolution Job in a specific domain
get_integration Returns an integration for a domain
get_matches Before calling this API, use CreateDomain or UpdateDomain to enable identity resolution
get_profile_object_type Returns the object types for a specific domain
datapipeline

get_profile_object_type_template
get_similar_profiles
get_workflow
get_workflow_steps
list_account_integrations
list_calculated_attribute_definitions
list_calculated_attributes_for_profile
list_domains
list_event_streams
list_identity_resolution_jobs
list_integrations
list_profile_objects
list_profile_object_type
list_profile_object_type_templates
list_rule_based_matches
list_tags_for_resource
list_workflows
merge_profiles
put_integration
put_profile_object
put_profile_object_type
search_profiles
tag_resource
untag_resource
update_calculated_attribute_definition
update_domain
update_profile

Returns the template information for a specific object type
Returns a set of profiles that belong to the same matching group using the matchId or profileId
Get details of specified workflow
Get granular list of steps in workflow
Lists all of the integrations associated to a specific URI in the AWS account
Lists calculated attribute definitions for Customer Profiles
Retrieve a list of calculated attributes for a customer profile
Returns a list of all the domains for an AWS account that have been created
Returns a list of all the event streams in a specific domain
Lists all of the Identity Resolution Jobs in your domain
Lists all of the integrations in your domain
Returns a list of objects associated with a profile of a given ProfileObjectType
Lists all of the templates available within the service
Lists all of the template information for object types
Returns a set of MatchIds that belong to the given domain
Displays the tags associated with an Amazon Connect Customer Profiles resource
Query to list all workflows
Runs an AWS Lambda job that does the following:
Adds an integration between the service and a third-party service, which includes Amazon AppFlow and Amazon Connect
Adds additional objects to customer profiles of a given ObjectType
Defines a ProfileObjectType
Searches for profiles within a specific domain using one or more predefined search keys
Assigns one or more tags (key-value pairs) to the specified Amazon Connect Customer Profiles resource
Removes one or more tags from the specified Amazon Connect Customer Profiles resource
Updates an existing calculated attribute definition
Updates the properties of a domain, including creating or selecting a dead letter queue
Updates the properties of a profile

Examples

```r
## Not run:
svc <- customerprofiles()
svc$add_profile_key(Foo = 123)

## End(Not run)
```

 datapipeline  
 AWS Data Pipeline
Description

AWS Data Pipeline configures and manages a data-driven workflow called a pipeline. AWS Data Pipeline handles the details of scheduling and ensuring that data dependencies are met so that your application can focus on processing the data.

AWS Data Pipeline provides a JAR implementation of a task runner called AWS Data Pipeline Task Runner. AWS Data Pipeline Task Runner provides logic for common data management scenarios, such as performing database queries and running data analysis using Amazon Elastic MapReduce (Amazon EMR). You can use AWS Data Pipeline Task Runner as your task runner, or you can write your own task runner to provide custom data management.

AWS Data Pipeline implements two main sets of functionality. Use the first set to create a pipeline and define data sources, schedules, dependencies, and the transforms to be performed on the data. Use the second set in your task runner application to receive the next task ready for processing. The logic for performing the task, such as querying the data, running data analysis, or converting the data from one format to another, is contained within the task runner. The task runner performs the task assigned to it by the web service, reporting progress to the web service as it does so. When the task is done, the task runner reports the final success or failure of the task to the web service.

Usage

datapipeline(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

config Optional configuration of credentials, endpoint, and/or region.

• credentials:
  • creds:
    • access_key_id: AWS access key ID
    • secret_access_key: AWS secret access key
    • session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

• endpoint: The complete URL to use for the constructed client.

• region: The AWS Region used in instantiating the client.

• close_connection: Immediately close all HTTP connections.

• timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

• s3_force_path_style: Set this to True to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials**
Optional credentials shorthand for the config parameter

• **creds**:  
  – **access_key_id**: AWS access key ID  
  – **secret_access_key**: AWS secret access key  
  – **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- datapipeline(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
```
Anonymous = "logical",
endpoint = "string",
region = "string"
)

Operations

activate_pipeline

Validates the specified pipeline and starts processing pipeline tasks

add_tags

Adds or modifies tags for the specified pipeline

create_pipeline

Creates a new, empty pipeline

deactivate_pipeline

Deactivates the specified running pipeline

delete_pipeline

Deletes a pipeline, its pipeline definition, and its run history

describe_objects

Gets the object definitions for a set of objects associated with the pipeline

describe_pipelines

Retrieves metadata about one or more pipelines

evaluate_expression

Task runners call EvaluateExpression to evaluate a string in the context of the specified object

get_pipeline_definition

Gets the definition of the specified pipeline

list_pipelines

Lists the pipeline identifiers for all active pipelines that you have permission to access

poll_for_task

Task runners call PollForTask to receive a task to perform from AWS Data Pipeline

put_pipeline_definition

Adds tasks, schedules, and preconditions to the specified pipeline

query_objects

Queries the specified pipeline for the names of objects that match the specified set of conditions

remove_tags

Removes existing tags from the specified pipeline

report_task_progress

Task runners call ReportTaskProgress when assigned a task to acknowledge that it has the task

report_task_runner_heartbeat

Task runners call ReportTaskRunnerHeartbeat every 15 minutes to indicate that they are operational

set_status

Requests that the status of the specified physical or logical pipeline objects be updated in the specified pipeline

set_task_status

Task runners call SetTaskStatus to notify AWS Data Pipeline that a task is completed and provide information about the final status

validate_pipeline_definition

Validates the specified pipeline definition to ensure that it is well formed and can be run without error

Examples

## Not run:
svc <- datapipeline()
svcactivate_pipeline(
  Foo = 123
)

## End(Not run)
**Description**

Amazon DataZone is a data management service that enables you to catalog, discover, govern, share, and analyze your data. With Amazon DataZone, you can share and access your data across accounts and supported regions. Amazon DataZone simplifies your experience across Amazon Web Services services, including, but not limited to, Amazon Redshift, Amazon Athena, Amazon Web Services Glue, and Amazon Web Services Lake Formation.

**Usage**

datazone(config = list(), credentials = list(), endpoint = NULL, region = NULL)

**Arguments**

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
    - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

- **endpoint**: Optional shorthand for complete URL to use for the constructed client.

- **region**: Optional shorthand for AWS Region used in instantiating the client.
Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- datazone(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `accept_predictions`: Accepts automatically generated business-friendly metadata for your Amazon DataZone assets
- `accept_subscription_request`: Accepts a subscription request to a specific asset
- `cancel_metadata_generation_run`: Cancels the metadata generation run
- `cancel_subscription`: Cancels the subscription to the specified asset
- `create_asset`: Creates an asset in Amazon DataZone catalog
- `create_asset_revision`: Creates a revision of the asset
- `create_asset_type`: Creates a custom asset type
- `create_data_source`: Creates an Amazon DataZone data source
<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>create_domain</td>
<td>Creates an Amazon DataZone domain</td>
</tr>
<tr>
<td>create_environment</td>
<td>Create an Amazon DataZone environment</td>
</tr>
<tr>
<td>create_environment_profile</td>
<td>Creates an Amazon DataZone environment profile</td>
</tr>
<tr>
<td>create_form_type</td>
<td>Creates an Amazon DataZone metadata form type</td>
</tr>
<tr>
<td>create_glossary</td>
<td>Creates a business glossary term</td>
</tr>
<tr>
<td>create_glossary_term</td>
<td>Creates a group profile in Amazon DataZone</td>
</tr>
<tr>
<td>create_group_profile</td>
<td>Publishes a listing (a record of an asset at a given time) or removes a listing from Amazon DataZone project</td>
</tr>
<tr>
<td>create_listing_change_set</td>
<td>Creates an Amazon DataZone project</td>
</tr>
<tr>
<td>create_project</td>
<td>Creates a project membership in Amazon DataZone</td>
</tr>
<tr>
<td>create_project_membership</td>
<td>Creates a subscription grant in Amazon DataZone</td>
</tr>
<tr>
<td>create_subscription_request</td>
<td>Creates a subscription request in Amazon DataZone</td>
</tr>
<tr>
<td>create_subscription_target</td>
<td>Creates a subscription target in Amazon DataZone</td>
</tr>
<tr>
<td>create_user_profile</td>
<td>Creates a user profile in Amazon DataZone</td>
</tr>
<tr>
<td>delete_asset</td>
<td>Deletes an asset in Amazon DataZone</td>
</tr>
<tr>
<td>delete_asset_type</td>
<td>Deletes an asset type in Amazon DataZone</td>
</tr>
<tr>
<td>delete_data_source</td>
<td>Deletes a data source in Amazon DataZone</td>
</tr>
<tr>
<td>delete_domain</td>
<td>Deletes an Amazon DataZone domain</td>
</tr>
<tr>
<td>delete_environment</td>
<td>Deletes the blueprint configuration in Amazon DataZone</td>
</tr>
<tr>
<td>delete_environment_blueprint_configuration</td>
<td>Deletes an environment profile in Amazon DataZone</td>
</tr>
<tr>
<td>delete_environment_profile</td>
<td>Deletes and metadata form type in Amazon DataZone</td>
</tr>
<tr>
<td>delete_form_type</td>
<td>Deletes a business glossary term in Amazon DataZone</td>
</tr>
<tr>
<td>delete_glossary</td>
<td>Deletes a business glossary term in Amazon DataZone</td>
</tr>
<tr>
<td>delete_glossary_term</td>
<td>Deletes a listing (a record of an asset at a given time)</td>
</tr>
<tr>
<td>delete_group_profile</td>
<td>Deletes a project in Amazon DataZone</td>
</tr>
<tr>
<td>delete_project</td>
<td>Deletes project membership in Amazon DataZone</td>
</tr>
<tr>
<td>delete_project_membership</td>
<td>Deletes a subscription grant in Amazon DataZone</td>
</tr>
<tr>
<td>delete_subscription_request</td>
<td>Deletes a subscription request in Amazon DataZone</td>
</tr>
<tr>
<td>delete_subscription_target</td>
<td>Deletes a subscription target in Amazon DataZone</td>
</tr>
<tr>
<td>delete_time_series_data_points</td>
<td>Deletes the specified time series form for the specified asset</td>
</tr>
<tr>
<td>get_asset</td>
<td>Gets an Amazon DataZone asset</td>
</tr>
<tr>
<td>get_asset_type</td>
<td>Gets an Amazon DataZone asset type</td>
</tr>
<tr>
<td>get_data_source</td>
<td>Gets an Amazon DataZone data source</td>
</tr>
<tr>
<td>get_data_source_run</td>
<td>Gets an Amazon DataZone data source run</td>
</tr>
<tr>
<td>get_domain</td>
<td>Gets an Amazon DataZone domain</td>
</tr>
<tr>
<td>get_environment</td>
<td>Gets an Amazon DataZone environment</td>
</tr>
<tr>
<td>get_environment_blueprint</td>
<td>Gets an Amazon DataZone blueprint</td>
</tr>
<tr>
<td>get_environment_blueprint_config</td>
<td>Gets the blueprint configuration in Amazon DataZone</td>
</tr>
<tr>
<td>get_environment_profile</td>
<td>Gets an environment profile in Amazon DataZone</td>
</tr>
<tr>
<td>get_form_type</td>
<td>Gets a metadata form type in Amazon DataZone</td>
</tr>
<tr>
<td>get_glossary</td>
<td>Gets a business glossary in Amazon DataZone</td>
</tr>
<tr>
<td>get_glossary_term</td>
<td>Gets a business glossary term in Amazon DataZone</td>
</tr>
<tr>
<td>get_group_profile</td>
<td>Gets a group profile in Amazon DataZone</td>
</tr>
<tr>
<td>get_iam_portal_login_url</td>
<td>Gets the data portal URL for the specified Amazon DataZone domain</td>
</tr>
<tr>
<td>get_listing</td>
<td>Gets a listing (a record of an asset at a given time)</td>
</tr>
<tr>
<td>get_metadata_generation_run</td>
<td>Gets a metadata generation run in Amazon DataZone</td>
</tr>
<tr>
<td>get_project</td>
<td>Gets a project in Amazon DataZone</td>
</tr>
</tbody>
</table>
get_subscription
get_subscription_grant
get_subscription_request_details
get_subscription_target
get_time_series_data_point
get_user_profile
list_asset_revisions
list_data_source_run_activities
list_data_source_runs
list_data_sources
list_domains
list_environment_blueprint_configurations
list_environment_blueprints
list_environment_profiles
list_environments
list_metadata_generation_runs
list_notifications
list_project_memberships
list_projects
list_subscription_grants
list_subscription_requests
list_subscriptions
list_subscription_targets
list_tags_for_resource
list_time_series_data_points
post_time_series_data_points
put_environment_blueprint_configuration
reject_predictions
reject_subscription_request
revoke_subscription
search
search_group_profiles
search_listings
search_types
search_user_profiles
start_data_source_run
start_metadata_generation_run
tag_resource
untag_resource
update_data_source
update_domain
update_environment
update_environment_profile
update_glossary
update_glossary_term
update_group_profile
update_project
update_subscription_grant_status

- Gets a subscription in Amazon DataZone
- Gets the subscription grant in Amazon DataZone
- Gets the details of the specified subscription request
- Gets the subscription target in Amazon DataZone
- Gets the existing data point for the asset
- Gets a user profile in Amazon DataZone
- Lists the revisions for the asset
- Lists data source run activities
- Lists data source runs in Amazon DataZone
- Lists data sources in Amazon DataZone
- Lists Amazon DataZone domains
- Lists blueprint configurations for a Amazon DataZone environment
- Lists blueprints in an Amazon DataZone environment
- Lists Amazon DataZone environment profiles
- Lists Amazon DataZone environments
- Lists all metadata generation runs
- Lists all Amazon DataZone notifications
- Lists all members of the specified project
- Lists Amazon DataZone projects
- Lists subscription grants
- Lists Amazon DataZone subscription requests
- Lists subscriptions in Amazon DataZone
- Lists subscription targets in Amazon DataZone
- Lists tags for the specified resource in Amazon DataZone
- Lists time series data points
- Posts time series data points to Amazon DataZone for the specified asset
- Writes the configuration for the specified environment blueprint in Amazon DataZone
- Rejects automatically generated business-friendly metadata for your Amazon DataZone assets
- Rejects the specified subscription request
- Revokes a specified subscription in Amazon DataZone
- Searches for assets in Amazon DataZone
- Searches group profiles in Amazon DataZone
- Searches listings (records of an asset at a given time) in Amazon DataZone
- Searches for types in Amazon DataZone
- Searches user profiles in Amazon DataZone
- Start the run of the specified data source in Amazon DataZone
- Starts the metadata generation run
- Tags a resource in Amazon DataZone
- Untags a resource in Amazon DataZone
- Updates the specified data source in Amazon DataZone
- Updates a Amazon DataZone domain
- Updates the specified environment in Amazon DataZone
- Updates the specified environment profile in Amazon DataZone
- Updates the business glossary in Amazon DataZone
- Updates a business glossary term in Amazon DataZone
- Updates the specified group profile in Amazon DataZone
- Updates the specified project in Amazon DataZone
- Updates the status of the specified subscription grant status in Amazon DataZone
update_subscription_request
update_subscription_target
update_user_profile

Updates a specified subscription request in Amazon DataZone
Updates the specified subscription target in Amazon DataZone
Updates the specified user profile in Amazon DataZone

Examples

```r
## Not run:
svc <- datazone()
svc$accept_predictions(
  Foo = 123
)
## End(Not run)
```

---

dax

Amazon DynamoDB Accelerator (DAX)

Description

DAX is a managed caching service engineered for Amazon DynamoDB. DAX dramatically speeds up database reads by caching frequently-accessed data from DynamoDB, so applications can access that data with sub-millisecond latency. You can create a DAX cluster easily, using the AWS Management Console. With a few simple modifications to your code, your application can begin taking advantage of the DAX cluster and realize significant improvements in read performance.

Usage

dax(config = list(), credentials = list(), endpoint = NULL, region = NULL)

Arguments

```r
class(dax)
```

* config: Optional configuration of credentials, endpoint, and/or region.
  * credentials:
    * creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    * profile: The name of a profile to use. If not given, then the default profile is used.
    * anonymous: Set anonymous credentials.
  * endpoint: The complete URL to use for the constructed client.
  * region: The AWS Region used in instantiating the client.
  * close_connection: Immediately close all HTTP connections.
• **timeout:** The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

• **s3_force_path_style:** Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

• **sts_regional_endpoint:** Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoint.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoint.html)

**credentials** Optional credentials shorthand for the config parameter

• **creds:**
  – **access_key_id**: AWS access key ID
  – **secret_access_key**: AWS secret access key
  – **session_token**: AWS temporary session token

• **profile:** The name of a profile to use. If not given, then the default profile is used.

• **anonymous:** Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- dax(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      region = "string",
      close_connection = "logical",
      timeout = "numeric",
      s3_force_path_style = "logical",
      sts_regional_endpoint = "string"
    ),
    credentials = list(
      creds = list(
        access_key_id = "string",
        region = "string",
        close_connection = "logical",
        timeout = "numeric",
        s3_force_path_style = "logical",
        sts_regional_endpoint = "string"
      )
    )
)```

secret_access_key = "string",
session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

create_cluster    Creates a DAX cluster
create_parameter_group  Creates a new parameter group
create_subnet_group   Creates a new subnet group
decrease_replication_factor  Removes one or more nodes from a DAX cluster
delete_cluster       Deletes a previously provisioned DAX cluster
delete_parameter_group  Deletes the specified parameter group
delete_subnet_group   Deletes a subnet group
describe_clusters    Returns information about all provisioned DAX clusters if no cluster identifier is specified, or about a specific DAX cluster if a cluster identifier is supplied
describe_default_parameters Returns the default system parameter information for the DAX caching software
describe_events      Returns events related to DAX clusters and parameter groups
describe_parameter_groups Returns a list of parameter group descriptions
describe_parameters   Returns the detailed parameter list for a particular parameter group
describe_subnet_groups Returns a list of subnet group descriptions
increase_replication_factor  Adds one or more nodes to a DAX cluster
list_tags            List all of the tags for a DAX cluster
reboot_node          Reboots a single node of a DAX cluster
tag_resource         Associates a set of tags with a DAX resource
untag_resource       Removes the association of tags from a DAX resource
update_cluster       Modifies the settings for a DAX cluster
update_parameter_group  Modifies the parameters of a parameter group
update_subnet_group   Modifies an existing subnet group

Examples

```r
## Not run:
svc <- dax()
svc$create_cluster(
   Foo = 123
)
## End(Not run)
```
Description

Detective uses machine learning and purpose-built visualizations to help you to analyze and investigate security issues across your Amazon Web Services (Amazon Web Services) workloads. Detective automatically extracts time-based events such as login attempts, API calls, and network traffic from CloudTrail and Amazon Virtual Private Cloud (Amazon VPC) flow logs. It also extracts findings detected by Amazon GuardDuty.

The Detective API primarily supports the creation and management of behavior graphs. A behavior graph contains the extracted data from a set of member accounts, and is created and managed by an administrator account.

To add a member account to the behavior graph, the administrator account sends an invitation to the account. When the account accepts the invitation, it becomes a member account in the behavior graph.

Detective is also integrated with Organizations. The organization management account designates the Detective administrator account for the organization. That account becomes the administrator account for the organization behavior graph. The Detective administrator account is also the delegated administrator account for Detective in Organizations.

The Detective administrator account can enable any organization account as a member account in the organization behavior graph. The organization accounts do not receive invitations. The Detective administrator account can also invite other accounts to the organization behavior graph.

Every behavior graph is specific to a Region. You can only use the API to manage behavior graphs that belong to the Region that is associated with the currently selected endpoint.

The administrator account for a behavior graph can use the Detective API to do the following:

- Enable and disable Detective. Enabling Detective creates a new behavior graph.
- View the list of member accounts in a behavior graph.
- Add member accounts to a behavior graph.
- Remove member accounts from a behavior graph.
- Apply tags to a behavior graph.

The organization management account can use the Detective API to select the delegated administrator for Detective.

The Detective administrator account for an organization can use the Detective API to do the following:

- Perform all of the functions of an administrator account.
- Determine whether to automatically enable new organization accounts as member accounts in the organization behavior graph.

An invited member account can use the Detective API to do the following:

- View the list of behavior graphs that they are invited to.
• Accept an invitation to contribute to a behavior graph.
• Decline an invitation to contribute to a behavior graph.
• Remove their account from a behavior graph.

All API actions are logged as CloudTrail events. See Logging Detective API Calls with CloudTrail.

We replaced the term "master account" with the term "administrator account". An administrator account is used to centrally manage multiple accounts. In the case of Detective, the administrator account manages the accounts in their behavior graph.

Usage

detective(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-encoding.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
profile: The name of a profile to use. If not given, then the default profile is used.

anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

svc <- detective(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)

Operations

accept_invitation Accepts an invitation for the member account to contribute data to a behavior graph
batch_get_graph_member_datasources
batch_get_membership_datasources
create_graph
create_members
delete_graph
delete_members
describe_organization_configuration
disable_organization_admin_account
dissociate_membership
disable_organization_admin_account
get_investigation
get_members
list_datasource_packages
list_graphs
list_indicators
list_investigations
list_invitations
list_members
list_organization_admin_accounts
list_tags_for_resource
reject_invitation
start_investigation
start_monitoring_member
tag_resource
untag_resource
update_datasource_packages
update_investigation_state
update_organization_configuration

Gets data source package information for the behavior graph
Gets information on the data source package history for an account
Creates a new behavior graph for the calling account, and sets that account as the administrator account
CreateMembers is used to send invitations to accounts
Disables the specified behavior graph and queues it to be deleted
Removes the specified member accounts from the behavior graph
Returns information about the configuration for the organization behavior graph
Removes the Detective administrator account in the current Region
Removes the member account from the specified behavior graph
Designates the Detective administrator account for the organization in the current Region
Detective investigations lets you investigate IAM users and IAM roles using indicators of compromise
Returns the membership details for specified member accounts for a behavior graph
Lists data source packages in the behavior graph
Returns the list of behavior graphs that the calling account is an administrator account of
Gets the indicators from an investigation
Detective investigations lets you investigate IAM users and IAM roles using indicators of compromise
Retrieves the list of open and accepted behavior graph invitations for the member account
Retrieves the list of member accounts for a behavior graph
Returns information about the Detective administrator account for an organization
Returns the tag values that are assigned to a behavior graph
Rejects an invitation to contribute the account data to a behavior graph
Detective investigations lets you investigate IAM users and IAM roles using indicators of compromise
Sends a request to enable data ingest for a member account that has a status of ACCEPTED_BUT_DISABLED
Applies tag values to a behavior graph
Removes tags from a behavior graph
Starts a data source packages for the behavior graph
Updates the state of an investigation
Updates the configuration for the Organizations integration in the current Region

Examples

```r
## Not run:
svc <- detective()
svc$accept_invitation(
    Foo = 123
)

## End(Not run)
```

---

Amazon DevOps Guru
**Description**

Amazon DevOps Guru is a fully managed service that helps you identify anomalous behavior in business critical operational applications. You specify the Amazon Web Services resources that you want DevOps Guru to cover, then the Amazon CloudWatch metrics and Amazon Web Services CloudTrail events related to those resources are analyzed. When anomalous behavior is detected, DevOps Guru creates an *insight* that includes recommendations, related events, and related metrics that can help you improve your operational applications. For more information, see [What is Amazon DevOps Guru](https://docs.aws.amazon.com/devopsguru/latest/ug/about-devops-guru.html).

You can specify 1 or 2 Amazon Simple Notification Service topics so you are notified every time a new insight is created. You can also enable DevOps Guru to generate an OpsItem in Amazon Web Services Systems Manager for each insight to help you manage and track your work addressing insights.

To learn about the DevOps Guru workflow, see [How DevOps Guru works](https://docs.aws.amazon.com/devopsguru/latest/ug/guru-workflow.html). To learn about DevOps Guru concepts, see [Concepts in DevOps Guru](https://docs.aws.amazon.com/devopsguru/latest/ug/concepts.html).

**Usage**

```r
devopsguru(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

**Arguments**

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)
credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value
A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- devopsguru(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string",
  close_connection = "logical",
  timeout = "numeric",
  s3_force_path_style = "logical",
  sts_regional_endpoint = "string"
)
```
devopsguru

    region = "string"
    )

Operations

add_notification_channel
delete_insight
describe_account_health
describe_account_overview
describe_anomaly
describe_event_sources_config
describe_feedback
describe_insight
describe_organization_health
describe_organization_overview
describe_organization_resource_collection_health
describe_resource_collection_health
describe_service_integration
get_cost_estimation
get_resource_collection
list_anomalies_for_insight
list_anomalous_log_groups
list_events
list_insights
list_monitored_resources
list_notification_channels
list_organization_insights
list_recommendations
put_feedback
remove_notification_channel
search_insights
search_organization_insights
start_cost_estimation
update_event_sources_config
update_resource_collection
update_service_integration

Adds a notification channel to DevOps Guru
Deletes the insight along with the associated anomalies, events and recommendations
Returns the number of open reactive insights, the number of open proactive insights, and the number of metrics analyzed in your Amazon Web Services account
For the time range passed in, returns the number of open reactive insights
Returns details about an anomaly that you specify using its ID
Returns the integration status of services that are integrated with DevOps Guru
Returns the most recent feedback submitted in the current Amazon Web Services account
Returns details about an insight that you specify using its ID
Returns active insights, predictive insights, and resource hours analyzed in the current Amazon Web Services account
Returns an overview of your organization’s history based on the specified time range
Provides an overview of your system’s health
Returns the number of open proactive insights, open reactive insights, and the number of metrics analyzed
Returns the integration status of services that are integrated with DevOps Guru
Returns an estimate of the monthly cost for DevOps Guru to analyze your Amazon Web Services resources
Returns lists Amazon Web Services resources that are of the specified resource collection type
Returns a list of the anomalies that belong to an insight that you specify using its ID
Returns the list of log groups that contain log anomalies
Returns a list of the events emitted by the resources that are evaluated by DevOps Guru
Returns a list of insights in your Amazon Web Services account
Returns the list of all log groups that are being monitored and tagged by DevOps Guru
Returns a list of notification channels configured for DevOps Guru
Returns a list of insights associated with the account or OU ID
Returns a list of a specified insight’s recommendations
Collects customer feedback about the specified insight
Removes a notification channel from DevOps Guru
Returns a list of insights in your Amazon Web Services account
Returns a list of insights in your organization
Starts the creation of an estimate of the monthly cost to analyze your Amazon Web Services resources
Enables or disables integration with a service that can be integrated with DevOps Guru
Updates the collection of resources that DevOps Guru analyzes
Enables or disables integration with a service that can be integrated with DevOps Guru

Examples

    ## Not run:
    svc <- devopsguru()
    svc$add_notification_channel(
        Foo = 123
    )

    ## End(Not run)
directconnect  AWS Direct Connect

Description

Direct Connect links your internal network to an Direct Connect location over a standard Ethernet fiber-optic cable. One end of the cable is connected to your router, the other to an Direct Connect router. With this connection in place, you can create virtual interfaces directly to the Amazon Web Services Cloud (for example, to Amazon EC2 and Amazon S3) and to Amazon VPC, bypassing Internet service providers in your network path. A connection provides access to all Amazon Web Services Regions except the China (Beijing) and (China) Ningxia Regions. Amazon Web Services resources in the China Regions can only be accessed through locations associated with those Regions.

Usage

directconnect(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

config   Optional configuration of credentials, endpoint, and/or region.

  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.

  • endpoint: The complete URL to use for the constructed client.

  • region: The AWS Region used in instantiating the client.

  • close_connection: Immediately close all HTTP connections.

  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoint.html

credentials   Optional credentials shorthand for the config parameter
directconnect

- **creds:**
  - `access_key_id`: AWS access key ID
  - `secret_access_key`: AWS secret access key
  - `session_token`: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**
A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- directconnect(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```
Operations

accept_direct_connect_gateway_associationProposal
allocate_connection_on_interconnect
allocate_hosted_connection
allocate_private_virtual_interface
allocate_public_virtual_interface
allocate_transit_virtual_interface
associate_connection_with_lag
associate_mac_sec_key
associate_virtual_interface
confirm_connection
confirm_customer_agreement
confirm_private_virtual_interface
confirm_public_virtual_interface
confirm_transit_virtual_interface
create_bgp_peer
create_connection
create_direct_connect_gateway
create_direct_connect_gateway_association
create_direct_connect_gateway_associationProposal
create_interconnect
create_lag
create_private_virtual_interface
create_transit_virtual_interface
delete_bgp_peer
delete_connection
delete_direct_connect_gateway
delete_direct_connect_gateway_association
delete_direct_connect_gateway_associationProposal
delete_interconnect
delete_lag
delete_virtual_interface
describe_connection_loa
describe_connections
describe_connections_on_interconnect
describe_customer_metadata
describe_direct_connect_gateway_associationProposals
describe_direct_connect_gateway_associations
describe_direct_connect_gateway_attachments
describe_direct_connect_gateways
describe_hosted_connections
describe_interconnect_loa
describe_interconnects
describe_lags
describe_loa

accept_direct_connect_gateway_associationProposal
Accepts a proposal request to attach a virtual private gateway or transit gateway to a Direct Connect gateway

allocate_connection_on_interconnect
Deprecated

allocate_hosted_connection
Creates a hosted connection on the specified interconnect or a link aggregation group (LAG)

allocate_private_virtual_interface
Provisions a private virtual interface to be owned by the specified Amazon Web Services account

allocate_public_virtual_interface
Provisions a public virtual interface to be owned by the specified Amazon Web Services account

allocate_transit_virtual_interface
Provisions a transit virtual interface to be owned by the specified Amazon Web Services account

associate_connection_with_lag
Associates an existing connection with a link aggregation group (LAG)

associate_mac_sec_key
Associates a MAC Security (MACsec) Connection Key Name (CKN)/Connectivity Association Key (CAK) pair with a Direct Connect dedicated connection

associate_virtual_interface
Associates a virtual interface with a specified link aggregation group (LAG) or connection

confirm_connection
Confirms the creation of the specified hosted connection on an interconnect

confirm_customer_agreement
The confirmation of the terms of agreement when creating the connection

confirm_private_virtual_interface
Accepts ownership of a private virtual interface created by another Amazon Web Services account

confirm_public_virtual_interface
Accepts ownership of a public virtual interface created by another Amazon Web Services account

confirm_transit_virtual_interface
Accepts ownership of a transit virtual interface created by another Amazon Web Services account

create_bgp_peer
Creates a BGP peer on the specified virtual interface

create_connection
Creates a connection between a customer network and a specific Direct Connect location

create_direct_connect_gateway
Creates a Direct Connect gateway, which is an intermediate object that enables you to connect a set of virtual interfaces and virtual private gateways

create_direct_connect_gateway_association
Creates an association between a Direct Connect gateway and a virtual private gateway or transit gateway

create_direct_connect_gateway_associationProposal
Creates a proposal to associate the specified virtual private gateway or transit gateway with the specified Direct Connect gateway

create_interconnect
Creates an interconnect between an Direct Connect Partner's network and a specific Direct Connect location

create_lag
Creates a link aggregation group (LAG) with the specified number of bundled physical dedicated connections between the customer network and a specific Direct Connect location

create_private_virtual_interface
Creates a private virtual interface

create_public_virtual_interface
Creates a public virtual interface

create_transit_virtual_interface
Creates a transit virtual interface

delete_bgp_peer
Deletes the specified BGP peer on the specified virtual interface with the specified customer address and ASN

delete_connection
Deletes the specified connection

delete_direct_connect_gateway
Deletes the specified Direct Connect gateway

delete_direct_connect_gateway_association
Deletes the association between the specified Direct Connect gateway and a virtual private gateway or transit gateway

delete_direct_connect_gateway_associationProposal
Deletes the association proposal request between the specified Direct Connect gateway and the specified virtual private gateway or transit gateway

delete_interconnect
Deletes the specified interconnect

delete_lag
Deletes the specified link aggregation group (LAG)

delete_virtual_interface
Deletes a virtual interface

describe_connection_loa
Deprecated

describe_connections
Get and view a list of customer agreements, along with their signed status and whether the customer is an NNIPartner, NNIPartnerV2, or a nonPartner

describe_connections_on_interconnect
Describes one or more association proposals for connection between direct connect gateway and virtual private gateway or transit gateway

describe_customer_metadata
Lists the associations between your Direct Connect gateways and virtual private gateways or transit gateways

describe_direct_connect_gateway_associationProposals
Lists the associations between your Direct Connect gateways and virtual private gateways or transit gateways

describe_direct_connect_gateway_associations
Lists the associations between your Direct Connect gateways and virtual private gateways or transit gateways

describe_direct_connect_gateway_attachments
Lists the attachments between your Direct Connect gateways and virtual private gateways or transit gateways

describe_direct_connect_gateways
Lists all your Direct Connect gateways or only the specified Direct Connect gateway

describe_hosted_connections
Lists the hosted connections that have been provisioned on the specified interconnect or link aggregation group

describe_interconnect_loa
Deprecated

describe_interconnects
Describes all your link aggregation groups (LAG) or the specified link aggregation group (LAG)

describe_lags
Get the LOA-CFA for a connection, interconnect, or link aggregation group

describe_loa
Deprecated
directoryservice

describe_locations
describe_router_configuration
describe_tags
describe_virtual_gateways
describe_virtual_interfaces
disassociate_connection_from_lag
disassociate_mac_sec_key
list_virtual_interface_test_history
start_bgp_failover_test
stop_bgp_failover_test
tag_resource
untag_resource
update_connection
update_direct_connect_gateway
update_direct_connect_gateway_association
update_lag
update_virtual_interface_attributes

Lists the Direct Connect locations in the current Amazon Web Services Region
Details about the router
Describes the tags associated with the specified Direct Connect resource
Lists the virtual private gateways owned by the Amazon Web Services account
Displays all virtual interfaces for an Amazon Web Services account
Disassociates a connection from a link aggregation group (LAG)
Removes the association between a MAC Security (MACsec) security key and a Direct Connect connection
Lists the virtual interface failover test history
Starts the virtual interface failover test that verifies your configuration meets your resiliency requirements
Stops the virtual interface failover test
Adds the specified tags to the specified Direct Connect resource
Removes one or more tags from the specified Direct Connect resource
Updates the Direct Connect dedicated connection configuration
Updates the name of a current Direct Connect gateway
Updates the specified attributes of the Direct Connect gateway association
Updates the attributes of the specified link aggregation group (LAG)
Updates the specified attributes of the specified virtual private interface

Examples

```r
## Not run:
svc <- directconnect()
svc$accept_direct_connect_gateway_association_proposal(
  Foo = 123
)

## End(Not run)
```

directoryservice  AWS Directory Service

Description

Directory Service

Directory Service is a web service that makes it easy for you to setup and run directories in the Amazon Web Services cloud, or connect your Amazon Web Services resources with an existing self-managed Microsoft Active Directory. This guide provides detailed information about Directory Service operations, data types, parameters, and errors. For information about Directory Services features, see Directory Service and the Directory Service Administration Guide.

Amazon Web Services provides SDKs that consist of libraries and sample code for various programming languages and platforms (Java, Ruby, .Net, iOS, Android, etc.). The SDKs provide a convenient way to create programmatic access to Directory Service and other Amazon Web Services services. For more information about the Amazon Web Services SDKs, including how to download and install them, see Tools for Amazon Web Services.
Usage

directoryservice(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

config  Optional configuration of credentials, endpoint, and/or region.

  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.

  • endpoint: The complete URL to use for the constructed client.

  • region: The AWS Region used in instantiating the client.

  • close_connection: Immediately close all HTTP connections.

  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials  Optional credentials shorthand for the config parameter

  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token

  • profile: The name of a profile to use. If not given, then the default profile is used.

  • anonymous: Set anonymous credentials.

endpoint  Optional shorthand for complete URL to use for the constructed client.

region  Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```r
svc <- directoryservice(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- **accept_shared_directory**: Accepts a directory sharing request that was sent from the directory owner account
- **add_ip_routes**: If the DNS server for your self-managed domain uses a publicly addressable IP address, you must add a CIDR address block to correctly route traffic to and from your Microsoft AD on Amazon Web Services
- **add_region**: Adds two domain controllers in the specified Region for the specified directory
- **add_tags_to_resource**: Adds or overwrites one or more tags for the specified directory
- **cancel_schema_extension**: Cancels an in-progress schema extension to a Microsoft AD directory
- **connect_directory**: Creates an AD Connector to connect to a self-managed directory
- **create_alias**: Creates an alias for a directory and assigns the alias to the directory
- **create_computer**: Creates an Active Directory computer object in the specified directory
- **create_conditional_forwarder**: Creates a conditional forwarder associated with your Amazon Web Services directory
- **create_directory**: Creates a Simple AD directory
- **create_log_subscription**: Creates a subscription to forward real-time Directory Service domain controller security logs to the specified Amazon CloudWatch log group
- **create_microsoft_ad**: Creates a Microsoft AD directory in the Amazon Web Services Cloud
- **create_snapshot**: Creates a snapshot of a Simple AD or Microsoft AD directory in the Amazon Web Services Cloud
- **create_trust**: Directory Service for Microsoft Active Directory allows you to configure trust relationships
directoryservice

- **delete_conditional_forwarder**: Deletes a conditional forwarder that has been set up for your Amazon Web Services directory.
- **delete_directory**: Deletes an Directory Service directory.
- **delete_log_subscription**: Deletes the specified log subscription.
- **delete_snapshot**: Deletes a directory snapshot.
- **delete_trust**: Deletes an existing trust relationship between your Managed Microsoft AD directory and an external domain.
- **deregister_certificate**: Deletes from the system the certificate that was registered for secure LDAP or client certificate authentication.
- **deregister_event_topic**: Removes the specified directory as a publisher to the specified Amazon SNS topic.
- **describe_certificate**: Displays information about the certificate registered for secure LDAP or client certificate authentication.
- **describe_client_authentication_settings**: Retrieves information about the type of client authentication for the specified directory.
- **describe_conditional_forwarders**: Obtains information about the conditional forwarders for this account.
- **describe_directories**: Obtains information about the directories that belong to this account.
- **describe_domain_controllers**: Provides information about any domain controllers in your directory.
- **describe_event_topics**: Obtains information about which Amazon SNS topics receive status messages from the specified directory.
- **describe_regions**: Provides information about the Regions that are configured for multi-Region replication.
- **describe_settings**: Retrieves information about the configurable settings for the specified directory.
- **describe_shared_directories**: Returns the shared directories in your account.
- **describe_snapshots**: Obtains information about the directory snapshots that belong to this account.
- **describe_trusts**: Obtains information about the trust relationships for this account.
- **describe_update_directory**: Describes the updates of a directory for a particular update type.
- **disable_client_authentication**: Disables alternative client authentication methods for the specified directory.
- **disable_ldaps**: Deactivates LDAP secure calls for the specified directory.
- **disable_radius**: Disables multi-factor authentication (MFA) with the Remote Authentication Dial In User Service (RADIUS) server for an AD Connector or Microsoft AD directory.
- **disable_sso**: Disables single-sign on for a directory.
- **enable_client_authentication**: Enables alternative client authentication methods for the specified directory.
- **enable_ldaps**: Activates the switch for the specific directory to always use LDAP secure calls.
- **enable_radius**: Enables multi-factor authentication (MFA) with the Remote Authentication Dial In User Service (RADIUS) server for an AD Connector or Microsoft AD directory.
- **enable_sso**: Enables single sign-on for a directory.
- **get_directory_limits**: Obtains directory limit information for the current Region.
- **get_snapshot_limits**: Obtains the manual snapshot limits for a directory.
- **list_certificates**: Lists all the certificates registered for a secure LDAP or client certificate authentication.
- **list_ip_routes**: Lists the address blocks that you have added to a directory.
- **list_log_subscriptions**: Lists the active log subscriptions for the Amazon Web Services account.
- **list_schema_extensions**: Lists all schema extensions applied to a Microsoft AD Directory.
- **list_tags_for_resource**: Lists all tags on a directory.
- **register_certificate**: Registers a certificate for a secure LDAP or client certificate authentication.
- **register_event_topic**: Associates a directory with an Amazon SNS topic.
- **reject_shared_directory**: Rejects a directory sharing request that was sent from the directory owner account.
- **remove_ip_routes**: Removes IP address blocks from a directory.
- **remove_region**: Stops all replication and removes the domain controllers from the specified Region.
- **remove_tags_from_resource**: Removes tags from a directory.
- **reset_user_password**: Resets the password for any user in your Managed Microsoft AD or Simple AD directory.
- **restore_from_snapshot**: Restores a directory using an existing directory snapshot.
- **share_directory**: Shares a specified directory (DirectoryId) in your Amazon Web Services account (directory consumer).
- **start_schema_extension**: Applies a schema extension to a Microsoft AD directory.
- **stop_schema_extension**: Removes a schema extension from a Microsoft AD directory.
- **unshare_directory**: Stops the directory sharing between the directory owner and consumer accounts.
- **update_conditional_forwarder**: Updates a conditional forwarder that has been set up for your Amazon Web Services directory.
- **update_directory_setup**: Updates the directory for a particular update type.
**update_number_of_domain_controllers**  Adds or removes domain controllers to or from the directory

**update_radius**  Updates the Remote Authentication Dial In User Service (RADIUS) server information for an AD Connector or Microsoft AD directory

**update_settings**  Updates the configurable settings for the specified directory

**update_trust**  Updates the trust that has been set up between your Managed Microsoft AD directory and an self-managed Active Directory

**verify_trust**  Directory Service for Microsoft Active Directory allows you to configure and verify

---

### Examples

```r
## Not run:
svc <- directoryservice()
svc$accept_shared_directory(
  Foo = 123
)

## End(Not run)
```

---

**dlm**  *Amazon Data Lifecycle Manager*

---

**Description**

With Amazon Data Lifecycle Manager, you can manage the lifecycle of your Amazon Web Services resources. You create lifecycle policies, which are used to automate operations on the specified resources.

Amazon Data Lifecycle Manager supports Amazon EBS volumes and snapshots. For information about using Amazon Data Lifecycle Manager with Amazon EBS, see Amazon Data Lifecycle Manager in the Amazon EC2 User Guide.

**Usage**

dlm(config = list(), credentials = list(), endpoint = NULL, region = NULL)

**Arguments**

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:  
    - **creds**:  
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
• **endpoint**: The complete URL to use for the constructed client.
• **region**: The AWS Region used in instantiating the client.
• **close_connection**: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends html

**credentials**

Optional credentials shorthand for the config parameter

• **creds**:  
  – **access_key_id**: AWS access key ID  
  – **secret_access_key**: AWS secret access key  
  – **session_token**: AWS temporary session token
• **profile**: The name of a profile to use. If not given, then the default profile is used.
• **anonymous**: Set anonymous credentials.

**endpoint**

Optional shorthand for complete URL to use for the constructed client.

**region**

Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- dlm(  
  config = list(  
    credentials = list(  
      creds = list(  
        access_key_id = "string",  
        secret_access_key = "string",  
        session_token = "string"  
      ),  
      profile = "string",  
      anonymous = "logical"  
    ),  
    endpoint = "string",  
    region = "string",  
    close_connection = "logical",  
    timeout = "numeric",  
    s3_force_path_style = "logical",  
    sts_regional_endpoint = "string"  
  )  
)
```
Amazon DocumentDB is a fast, reliable, and fully managed database service. Amazon DocumentDB makes it easy to set up, operate, and scale MongoDB-compatible databases in the cloud. With Amazon DocumentDB, you can run the same application code and use the same drivers and tools that you use with MongoDB.
Usage

docdb(config = list(), credentials = list(), endpoint = NULL, region = NULL)

Arguments

cfg Optional configuration of credentials, endpoint, and/or region.

- credentials:
  - creds:
    - access_key_id: AWS access key ID
    - secret_access_key: AWS secret access key
    - session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.

- endpoint: The complete URL to use for the constructed client.
- region: The AWS Region used in instantiating the client.
- close_connection: Immediately close all HTTP connections.
- timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

- sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

creds Optional credentials shorthand for the config parameter

- access_key_id: AWS access key ID
- secret_access_key: AWS secret access key
- session_token: AWS temporary session token
- profile: The name of a profile to use. If not given, then the default profile is used.
- anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

svc <- docdb(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)

Operations

add_source_identifier_to_subscription Adds a source identifier to an existing event notification subscription
add_tags_to_resource Adds metadata tags to an Amazon DocumentDB resource
apply_pending_maintenance_action Applies a pending maintenance action to a resource (for example, to an Amazon DocumentDB instance)
copy_db_cluster_parameter_group Copies the specified cluster parameter group
copy_db_cluster_snapshot Copies a snapshot of a cluster
copy_db_cluster Creates a new Amazon DocumentDB cluster
copy_db_cluster_parameter_group Creates a new cluster parameter group
copy_db_cluster_snapshot Creates a snapshot of a cluster
copy_db_instance Creates a new instance
copy_db_subnet_group Creates a new subnet group
copy_event_subscription Creates an Amazon DocumentDB event notification subscription
copy_global_cluster Creates an Amazon DocumentDB global cluster that can span multiple multiple AWS Regions
delete_db_cluster Deletes a previously provisioned cluster
delete_db_cluster_parameter_group Deletes a specified cluster parameter group
delete_db_cluster_snapshot
delete_db_instance
delete_db_subnet_group
delete_event_subscription
delete_global_cluster
describe_certificates
describe_db_cluster_parameter_groups
describe_db_cluster_parameters
describe_db_clusters
describe_db_cluster_snapshot_attributes
describe_db_cluster_snapshots
describe_db_engine_versions
describe_db.instances
describe_db_subnet_groups
describe_engine_default_cluster_parameters
describe_event_categories
describe_events
describe_event_subscriptions
describe_global_clusters
describe_orderable_db_instance_options
describe_pending_maintenance_actions
failover_db_cluster
list_tags_for_resource
modify_db_cluster
modify_db_cluster_parameter_group
modify_db_cluster_snapshot_attribute
modify_db_instance
modify_db_subnet_group
modify_event_subscription
modify_global_cluster
reboot_db_instance
remove_from_global_cluster
remove_source_identifier_from_subscription
remove_tags_from_resource
reset_db_cluster_parameter_group
restore_db_cluster_from_snapshot
restore_db_cluster_to_point_in_time
start_db_cluster
stop_db_cluster
switchover_global_cluster

Deletes a cluster snapshot
Deletes a previously provisioned instance
Deletes a subnet group
Deletes an Amazon DocumentDB event notification subscription
Deletes a global cluster
Returns a list of certificate authority (CA) certificates provided by Amazon DocumentDB
Returns a list of DBClusterParameterGroup descriptions
Returns the detailed parameter list for a particular cluster parameter group
Returns information about provisioned Amazon DocumentDB clusters
Returns a list of cluster snapshot attribute names and values for a manual DB cluster snapshot
Returns information about cluster snapshots
Returns a list of the available engines
Returns information about provisioned Amazon DocumentDB instances
Returns a list of DBSubnetGroup descriptions
Returns the default engine and system parameter information for the cluster database engine
Displays a list of categories for all event source types, or, if specified, for a specific source type
Returns events related to instances, security groups, snapshots, and DB parameter groups
Lists all the subscription descriptions for a customer account
Returns information about Amazon DocumentDB global clusters
Returns a list of orderable instance options for the specified engine
Returns a list of resources (for example, instances) that have at least one pending maintenance action
Forces a failover for a cluster
Lists all tags on an Amazon DocumentDB resource
Modifies a setting for an Amazon DocumentDB cluster
Modifies the parameters of a cluster parameter group
Adds an attribute and values to, or removes an attribute and values from, a specified cluster parameter group
Modifies settings for an instance
Modifies an existing subnet group
Modifies an existing Amazon DocumentDB event notification subscription
Modify a setting for an Amazon DocumentDB global cluster
You might need to reboot your instance, usually for maintenance reasons
Detaches an Amazon DocumentDB secondary cluster from a global cluster
Removes a source identifier from an existing Amazon DocumentDB event notification subscription
Removes metadata tags from an Amazon DocumentDB resource
Modifies the parameters of a cluster parameter group to the default value
Creates a new cluster from a snapshot or cluster snapshot
Restores a cluster to an arbitrary point in time
Starts the stopped cluster that is specified by DBClusterIdentifier
Stops the running cluster that is specified by DBClusterIdentifier
Switches over the specified secondary Amazon DocumentDB cluster to be the

Examples

```r
## Not run:
svc <- docdb()
svc$add_source_identifier_to_subscription(Foo = 123)
```
Amazon DocumentDB Elastic Clusters

Description

Amazon DocumentDB elastic clusters support workloads with millions of reads/writes per second and petabytes of storage capacity. Amazon DocumentDB elastic clusters also simplify how developers interact with Amazon DocumentDB elastic-clusters by eliminating the need to choose, manage or upgrade instances.

Amazon DocumentDB elastic-clusters were created to:

• provide a solution for customers looking for a database that provides virtually limitless scale with rich query capabilities and MongoDB API compatibility.
• give customers higher connection limits, and to reduce downtime from patching.
• continue investing in a cloud-native, elastic, and class leading architecture for JSON workloads.

Usage

docdbelastic(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)

Arguments

Optional configuration of credentials, endpoint, and/or region.

• **config**: Optional configuration of credentials, endpoint, and/or region.
  
  • **credentials**: Optional credentials configuration.
    
    • **creds**:
      
      * **access_key_id**: AWS access key ID
      * **secret_access_key**: AWS secret access key
      * **session_token**: AWS temporary session token
    
    • **profile**: The name of a profile to use. If not given, then the default profile is used.
    
    • **anonymous**: Set anonymous credentials.
  
  • **endpoint**: The complete URL to use for the constructed client.
  
  • **region**: The AWS Region used in instantiating the client.
• **close_connection**: Immediately close all HTTP connections.

• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

• **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.

• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature sts-regionalized-](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials** Optional credentials shorthand for the config parameter

• **creds**:
  – **access_key_id**: AWS access key ID
  – **secret_access_key**: AWS secret access key
  – **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- docdbelastic(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
```
creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

- `copy_cluster_snapshot` Copies a snapshot of an elastic cluster
- `create_cluster` Creates a new Amazon DocumentDB elastic cluster and returns its cluster structure
- `create_cluster_snapshot` Creates a snapshot of an elastic cluster
- `delete_cluster` Delete an elastic cluster
- `delete_cluster_snapshot` Delete an elastic cluster snapshot
- `get_cluster` Returns information about a specific elastic cluster
- `get_cluster_snapshot` Returns information about a specific elastic cluster snapshot
- `list_clusters` Returns information about provisioned Amazon DocumentDB elastic clusters
- `list_cluster_snapshots` Returns information about snapshots for a specified elastic cluster
- `list_tags_for_resource` Lists all tags on a elastic cluster resource
- `restore_cluster_from_snapshot` Restores an elastic cluster from a snapshot
- `start_cluster` Restarts the stopped elastic cluster that is specified by clusterARN
- `stop_cluster` Stops the running elastic cluster that is specified by clusterArn
- `tag_resource` Adds metadata tags to an elastic cluster resource
- `untag_resource` Removes metadata tags from an elastic cluster resource
- `update_cluster` Modifies an elastic cluster

Examples

```r
## Not run:
svc <- docdbelastic()
svc$copy_cluster_snapshot(
    Foo = 123
)
## End(Not run)
```
Description

AWS Elastic Disaster Recovery Service.

Usage

drs(config = list(), credentials = list(), endpoint = NULL, region = NULL)

Arguments

config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      • access_key_id: AWS access key ID
      • secret_access_key: AWS secret access key
      • session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```r
svc <- drs(
  config = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string",
  close_connection = "logical",
  timeout = "numeric",
  s3_force_path_style = "logical",
  sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `associate_source_network_stack`: Associate a Source Network to an existing CloudFormation Stack and modifies launch templates to use this network.
- `create_extended_source_server`: Create an extended source server in the target Account based on the source server in staging account.
- `create_launch_configuration_template`: Creates a new Launch Configuration Template.
- `create_replication_configuration_template`: Creates a new ReplicationConfigurationTemplate.
- `create_source_network`: Create a new Source Network resource for a provided VPC ID.
- `delete_job`: Deletes a single Job by ID.
- `delete_launch_action`: Deletes a resource launch action.
- `delete_launch_configuration_template`: Deletes a single Launch Configuration Template by ID.
- `delete_recovery_instance`: Deletes a single Recovery Instance by ID.
- `delete_replication_configuration_template`: Deletes a single Replication Configuration Template by ID.
- `delete_source_network`: Delete Source Network resource.
- `delete_source_server`: Deletes a single Source Server by ID.
- `describe_job_log_items`: Retrieves a detailed Job log with pagination.
- `describe_jobs`: Returns a list of Jobs.
describe_launch_configuration_templates  
Lists all Launch Configuration Templates, filtered by Launch Configuration Template IDs

describe_recovery_instances  
Lists all Recovery Instances or multiple Recovery Instances by ID

describe_recovery_snapshots  
Lists all Recovery Snapshots for a single Source Server

describe_replication_configuration_templates  
Lists all ReplicationConfigurationTemplates, filtered by Source Server IDs

describe_source_networks  
Lists all Source Networks or multiple Source Networks filtered by ID

describe_source_servers  
Lists all Source Servers or multiple Source Servers filtered by ID

disconnect_recovery_instance  
Disconnect a Recovery Instance from Elastic Disaster Recovery

disconnect_source_server  
Disconnects a specific Source Server from Elastic Disaster Recovery

disconnect_source_server  
Export the Source Network CloudFormation template to an S3 bucket

describe_launch_configuration  
Lists all Failback ReplicationConfigurations, filtered by Recovery Instance ID

describe_replication_configuration  
Gets a LaunchConfiguration, filtered by Source Server IDs

describe_replication_configuration  
Gets a ReplicationConfiguration, filtered by Source Server ID

initialize_service  
Initialize Elastic Disaster Recovery

list_extensible_source_servers  
Returns a list of source servers on a staging account that are extensible, which
list_launch_actions  
Lists resource launch actions
list_source_networks  
Returns an array of staging accounts for existing extended source servers
list_staging_accounts  
List all tags for your Elastic Disaster Recovery resources
list_tags_for_resource  
Put a resource launch action

put_launch_action  
WARNING: RetryDataReplication is deprecated
retry_data_replication  
Start replication to origin / target region - applies only to protected instances that
reverse_replication  
Initiates a Job for launching the machine that is being failed back to from the
start_failback_launch  
Starts replication for a stopped Source Server
start_recovery  
Starts replication for a specified Recovery Instance
start_replication  
Starts replication for a Source Network
start_source_network_recovery  
Stops the failback process for a specified Recovery Instance
start_source_network_recovery  
Stops replication for a Source Server
start_source_network_replication  
Adds or overwrites only the specified tags for the specified Elastic Disaster Recovery
stop_failback  
Initiates a Job for terminating the EC2 resources associated with the specified
stop_replication  
tag_resource  
Deletes the specified set of tags from the specified set of Elastic Disaster Recovery
untag_resource  
Updates a LaunchConfiguration by Source Server ID
update_failback_replication_configuration  
Updates an existing LaunchConfigurationTemplate by ID
update_launch_configuration  
Updates an existing Launch Configuration Template by ID
update_replication_configuration  
Registrations
update_replication_configuration_template  
Updates a ReplicationConfigurationTemplate by ID

Examples

```r
## Not run:
svc <- drs()
svc$associate_source_network_stack(
  Foo = 123
)
## End(Not run)
```
Amazon DynamoDB

Description

Amazon DynamoDB is a fully managed NoSQL database service that provides fast and predictable performance with seamless scalability. DynamoDB lets you offload the administrative burdens of operating and scaling a distributed database, so that you don’t have to worry about hardware provisioning, setup and configuration, replication, software patching, or cluster scaling.

With DynamoDB, you can create database tables that can store and retrieve any amount of data, and serve any level of request traffic. You can scale up or scale down your tables' throughput capacity without downtime or performance degradation, and use the Amazon Web Services Management Console to monitor resource utilization and performance metrics.

DynamoDB automatically spreads the data and traffic for your tables over a sufficient number of servers to handle your throughput and storage requirements, while maintaining consistent and fast performance. All of your data is stored on solid state disks (SSDs) and automatically replicated across multiple Availability Zones in an Amazon Web Services Region, providing built-in high availability and data durability.

Usage

dynamodb(config = list(), credentials = list(), endpoint = NULL, region = NULL)

Arguments

config  Optional configuration of credentials, endpoint, and/or region.

• credentials:
  – creds:
    * access_key_id: AWS access key ID
    * secret_access_key: AWS secret access key
    * session_token: AWS temporary session token
  – profile: The name of a profile to use. If not given, then the default profile is used.
  – anonymous: Set anonymous credentials.

• endpoint: The complete URL to use for the constructed client.

• region: The AWS Region used in instantiating the client.

• close_connection: Immediately close all HTTP connections.

• timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

• s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

• sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html
credentials  Optional credentials shorthand for the config parameter

• creds:
  – access_key_id: AWS access key ID
  – secret_access_key: AWS secret access key
  – session_token: AWS temporary session token

• profile: The name of a profile to use. If not given, then the default profile is used.

• anonymous: Set anonymous credentials.

endpoint  Optional shorthand for complete URL to use for the constructed client.

region  Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- dynamodb(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
```
region = “string”
)

Operations

batch_execute_statement
batch_get_item
batch_write_item
create_backup
create_global_table
create_table
delete_backup
delete_item
delete_resource_policy
delete_table
describe_backup
describe_continuous_backups
describe_contributor_insights
describe_endpoints
describe_export
describe_global_table
describe_global_table_settings
describe_import
describe_kinesis_streaming_destination
describe_limits
describe_table
describe_table_replica_auto_scaling
describe_time_to_live
disable_kinesis_streaming_destination
enable_kinesis_streaming_destination
execute_statement
execute_transaction
export_table_to_point_in_time
get_item
get_resource_policy
import_table
list_backups
list_contributor_insights
list_exports
list_global_tables
list_imports
list_tables
list_tags_of_resource
put_item
put_resource_policy
query
restore_table_from_backup
restore_table_to_point_in_time

This operation allows you to perform batch reads or writes on data stored in DynamoDB. The BatchGetItem operation returns the attributes of one or more items from one or more tables. The BatchWriteItem operation puts or deletes multiple items in one or more tables. Creates a backup for an existing table. Creates a global table from an existing table. The CreateTable operation adds a new table to your account. Deletes an existing backup of a table. Deletes a single item in a table by primary key. Deletes the resource-based policy attached to the resource, which can be a table or stream. The DeleteTable operation deletes a table and all of its items. Describes an existing backup of a table. Checks the status of continuous backups and point in time recovery on the specified table. Returns information about contributor insights for a given table or global secondary index. Returns the regional endpoint information. Describes an existing table export. Returns information about the specified global table. Describes Region-specific settings for a global table. Represents the properties of the import. Returns information about the status of Kinesis streaming. Returns the current provisioned-capacity quotas for your Amazon Web Services account. Returns information about the table, including the current status of the table, when it was created, the primary key schema, and any indexes on the table. Describes auto scaling settings across replicas of the global table at once. Gives a description of the Time to Live (TTL) status on the specified table. Stops replication from the DynamoDB table to the Kinesis data stream. Starts table data replication to the specified Kinesis data stream at a timestamp chosen during the enable workflow. This operation allows you to perform reads and singleton writes on data stored in DynamoDB. This operation allows you to perform transactional reads or writes on data stored in DynamoDB. The GetItem operation returns a set of attributes for the item with the given primary key. Returns the resource-based policy document attached to the resource, which can be a table or stream. Imports table data from an S3 bucket. Lists DynamoDB backups that are associated with an Amazon Web Services account. Returns a list of ContributorInsightsSummary for a table and all its global secondary indexes. Lists completed exports within the past 90 days. Lists all global tables that have a replica in the specified Region. Lists completed imports within the past 90 days. Returns an array of table names associated with the current account and endpoint. List all tags on an Amazon DynamoDB resource. Creates a new item, or replaces an old item with a new item. Attaches a resource-based policy document to the resource, which can be a table or stream. You must provide the name of the partition key attribute and a single value for that attribute. Creates a new table from an existing backup. Restores the specified table to the specified point in time within EarliestRestorableDateTime.
scan
tag_resource
transact_get_items
transact_write_items
untag_resource
update_continuous_backups
update_contributor_insights
update_global_table
update_global_table_settings
update_item
update_kinesis_streaming_destination
update_table
update_table_replica_auto_scaling
update_time_to_live

The Scan operation returns one or more items and item attributes by accessing every item in a table or a secondary index.

Associate a set of tags with an Amazon DynamoDB resource.

TransactGetItems is a synchronous operation that atomically retrieves multiple items.

TransactWriteItems is a synchronous write operation that groups up to 100 action requests.

Removes the association of tags from an Amazon DynamoDB resource.

UpdateContinuousBackups enables or disables point in time recovery for the specified table.

Updates the status for contributor insights for a specific table or index.

Adds or removes replicas in the specified global table.

Updates settings for a global table.

Edits an existing item’s attributes, or adds a new item to the table if it does not already exist.

The command to update the Kinesis stream destination.

Modifies the provisioned throughput settings, global secondary indexes, or DynamoDB Streams settings for a given table.

Updates auto scaling settings on your global tables at once.

The UpdateTimeToLive method enables or disables Time to Live (TTL) for the specified table.

Examples

```r
## Not run:
svc <- dynamodb()

# This example reads multiple items from the Music table using a batch of
# three GetItem requests. Only the AlbumTitle attribute is returned.
svc$batch_get_item(
  RequestItems = list(
    Music = list(
      Keys = list(
        list(
          Artist = list(
            S = "No One You Know"
          ),
          SongTitle = list(
            S = "Call Me Today"
          )
        ),
        list(
          Artist = list(
            S = "Acme Band"
          ),
          SongTitle = list(
            S = "Happy Day"
          )
        ),
        list(
          Artist = list(
            S = "No One You Know"
          ),
          SongTitle = list(
            S = "Scared of My Shadow"
          )
        )
      )
    )
  )
)
```
### Amazon DynamoDB Streams

**Description**

Amazon DynamoDB

Amazon DynamoDB Streams provides API actions for accessing streams and processing stream records. To learn more about application development with Streams, see Capturing Table Activity with DynamoDB Streams in the Amazon DynamoDB Developer Guide.

**Usage**

```r
dynamodbstreams(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

**Arguments**

- `config`: Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
    - `creds`:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
    - `anonymous`: Set anonymous credentials.
  - `endpoint`: The complete URL to use for the constructed client.
  - `region`: The AWS Region used in instantiating the client.
  - `close_connection`: Immediately close all HTTP connections.
  - `timeout`: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - `s3_force_path_style`: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials** Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.

- **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- dynamodbstreams(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
```
anonymous = "logical"
endpoinl = "string",
region = "string"
}

Operations

describe_stream Returns information about a stream, including the current status of the stream, its Amazon Resource Name (ARN), the composition of its shards, and its corresponding DynamoDB table
get_records Retrieves the stream records from a given shard
get_shard_iterator Returns a shard iterator
list_streams Returns an array of stream ARNs associated with the current account and endpoint

Examples

## Not run:
svc <- dynamodbstreams()
# The following example describes a stream with a given stream ARN.
svc$describe_stream(
)
## End(Not run)

---

**Amazon Elastic Block Store**

**Description**

You can use the Amazon Elastic Block Store (Amazon EBS) direct APIs to create Amazon EBS snapshots, write data directly to your snapshots, read data on your snapshots, and identify the differences or changes between two snapshots. If you’re an independent software vendor (ISV) who offers backup services for Amazon EBS, the EBS direct APIs make it more efficient and cost-effective to track incremental changes on your Amazon EBS volumes through snapshots. This can be done without having to create new volumes from snapshots, and then use Amazon Elastic Compute Cloud (Amazon EC2) instances to compare the differences.

You can create incremental snapshots directly from data on-premises into volumes and the cloud to use for quick disaster recovery. With the ability to write and read snapshots, you can write your on-premises data to an snapshot during a disaster. Then after recovery, you can restore it back to Amazon Web Services or on-premises from the snapshot. You no longer need to build and maintain complex mechanisms to copy data to and from Amazon EBS.

This API reference provides detailed information about the actions, data types, parameters, and errors of the EBS direct APIs. For more information about the elements that make up the EBS
direct APIs, and examples of how to use them effectively, see Accessing the Contents of an Amazon EBS Snapshot in the Amazon Elastic Compute Cloud User Guide. For more information about the supported Amazon Web Services Regions, endpoints, and service quotas for the EBS direct APIs, see Amazon Elastic Block Store Endpoints and Quotas in the Amazon Web Services General Reference.

Usage

```r
ebs(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>config</code></td>
<td>Optional configuration of credentials, endpoint, and/or region.</td>
</tr>
</tbody>
</table>
|           | • `credentials`:
|           |   • `creds`:
|           |     • `access_key_id`: AWS access key ID
|           |     • `secret_access_key`: AWS secret access key
|           |     • `session_token`: AWS temporary session token
|           |   • `profile`: The name of a profile to use. If not given, then the default profile is used.
|           |   • `anonymous`: Set anonymous credentials.
|           | • `endpoint`: The complete URL to use for the constructed client.
|           | • `region`: The AWS Region used in instantiating the client.
|           | • `close_connection`: Immediately close all HTTP connections.
|           | • `timeout`: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
|           | • `s3_force_path_style`: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
|           | • `sts_regional_endpoint`: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)
| `credentials` | Optional credentials shorthand for the `config` parameter
|             | • `creds`:
|             |   • `access_key_id`: AWS access key ID
|             |   • `secret_access_key`: AWS secret access key
|             |   • `session_token`: AWS temporary session token
|             | • `profile`: The name of a profile to use. If not given, then the default profile is used.
|             | • `anonymous`: Set anonymous credentials.
| `endpoint`  | Optional shorthand for complete URL to use for the constructed client.
| `region`    | Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service's operations using syntax like `svc$operation(...)`, where `svc` is the name you've assigned to the client. The available operations are listed in the Operations section.
Service syntax

```r
svc <- ebs(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- **complete_snapshot** Seals and completes the snapshot after all of the required blocks of data have been written to it
- **get_snapshot_block** Returns the data in a block in an Amazon Elastic Block Store snapshot
- **list_changed_blocks** Returns information about the blocks that are different between two Amazon Elastic Block Store snapshots
- **list_snapshot_blocks** Returns information about the blocks in an Amazon Elastic Block Store snapshot
- **put_snapshot_block** Writes a block of data to a snapshot
- **start_snapshot** Creates a new Amazon EBS snapshot

Examples

```r
## Not run:
svc <- ebs()
svc$complete_snapshot()
```
Foo = 123
)

## End(Not run)

---

**ec2**  
*Amazon Elastic Compute Cloud*

---

**Description**

You can access the features of Amazon Elastic Compute Cloud (Amazon EC2) programmatically. For more information, see the [Amazon EC2 Developer Guide](https://docs.aws.amazon.com/DeveloperGuide/ec2).  

**Usage**

```r
ec2(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

**Arguments**

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:  
    - **creds**:  
      - access_key_id: AWS access key ID  
      - secret_access_key: AWS secret access key  
      - session_token: AWS temporary session token  
    - **profile**: The name of a profile to use. If not given, then the default profile is used.  
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.  
  - **region**: The AWS Region used in instantiating the client.  
  - **close_connection**: Immediately close all HTTP connections.  
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkforlatestguide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkforlatestguide/feature-sts-regionalized-ends.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:  
    - access_key_id: AWS access key ID  
    - secret_access_key: AWS secret access key  
    - session_token: AWS temporary session token
• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**
A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**
```
svc <- ec2(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string",
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

**Operations**

`accept_address_transfer`
Accepts an Elastic IP address transfer
accept_reserved_instances_exchange_quote
accept_transit_gateway_multicast_domain_associations
accept_transit_gateway_peering_attachment
accept_transit_gateway_vpc_attachment
accept_vpc_endpoint_connections
accept_vpc_peering_connection
advertise_byop_cidr
allocate_address
allocate_hosts
allocate_ipam_pool_cidr
apply_security_groups_to_client_vpn_target_network
assign_ipv_6_addresses
assign_private_ip_addresses
assign_private_nat_gateway_address
associate_address
associate_client_vpn_target_network
associate_dhcp_options
associate_enclave_certificate_iam_role
associate_iam_instance_profile
associate_instance_event_window
associate_ipam_byoasn
associate_ipam_resource_discovery
associate_nat_gateway_address
associate_route_table
associate_subnet_cidr_block
associate_transit_gateway_multicast_domain
associate_transit_gateway_policy_table
associate_transit_gateway_route_table
associate_trunk_interface
associate_vpc_cidr_block
attach_classic_link_vpc
attach_internet_gateway
attach_network_interface
attach_verified_access_trust_provider
attach_volume
attach_vpn_gateway
authorize_client_vpn_ingress
authorize_security_group_egress
authorize_security_group_ingress
bundle_instance
cancel_bundle_task
cancel_capacity_reservation
cancel_capacity_reservation_fleets
cancel_conversion_task
cancel_export_task
cancel_image_launch_permission
cancel_import_task
cancel_reserved_instances_listing

Accepts the Convertible Reserved Instance exchange quote
Accepts a request to associate subnets with a transit gateway
Accepts a transit gateway peering attachment request
Accepts a request to attach a VPC to a transit gateway
Accepts connection requests to your VPC endpoint
Accepts a VPC peering connection request
Advertises an IPv4 or IPv6 address range through bring your own IP addresses (BYOIP)
Allocates a Dedicated Host to your account
Allocate a CIDR from an IPAM pool
Applies a security group to the association between the target network and a Client VPN endpoint
Assigns one or more IPv6 addresses to the specified network interface
Assigns one or more secondary private IP addresses to a private IP address
Associates an Elastic IP address with a carrier IP address
Associates a target network with a Client VPN endpoint
Associates a set of DHCP options (that you’ve previously associated) with your VPC
Associates an Identity and Access Management (IAM) instance profile with a running or stopped instance
Associates your Autonomous System Number (ASN) with a BYOIP CIDR that you own
Associates Elastic IP addresses (EIPs) and private IPv4 addresses with a public NAT gateway
Assigns one or more IPv6 addresses to a private NAT gateway
Assigns one or more secondary private IP addresses to a private IP address
Associates a subnet with an internet gateway or virtual private gateway
Associates a subnet in your VPC or an internet gateway with a specified route table
Associates Device AWS One Zone IPv6 CIDR block with your subnet
Assigns one or more IPv6 addresses to the specified network interface
Associates the specified subnets and transit gateways
Associates the specified transit gateway with the specified attachment
Associates the specified attachment with the specified transit gateway
Associates a branch network interface with a trunk network interface
Associates a CIDR block with your VPC
This action is deprecated
Attaches an internet gateway or a virtual private gateway to your VPC
Attaches a network interface to an instance
Attaches the specified Amazon Web Services Verified Access trust provider
Attaches an Amazon Web Services Verified Access trust provider to your VPC
Attaches an EBS volume to a running or stopped instance
Attaches a virtual private gateway to a VPC
Adds an ingress authorization rule to a Client VPN endpoint
Add the specified outbound (egress) rules to a security group
Add the specified inbound (ingress) rules to a security group
Bundles an Amazon instance store-backed Windows image
Cancels a bundling operation for an instance store-backed Windows image
Cancels the specified Capacity Reservation, release all resources
Cancels one or more Capacity Reservation Fleets
Cancels an active conversion task
Cancels an active export task
Removes your Amazon Web Services account from the list of trusted accounts
Cancels an in-process import virtual machine or snapshot task
Cancels the specified Reserved Instance listing in the Reserved Instance Marketplace
cancel_spot_fleet_requests
cancel_spot_instance_requests
confirm_product_instance
copy_fpga_image
copy_image
copy_snapshot
create_capacity_reservation
create_capacity_reservation_fleet
create_carrier_gateway
create_client_vpn_endpoint
create_client_vpn_route
create_coip_cidr
create_coip_pool
create_customer_gateway
create_default_subnet
create_default_vpc
create_dhcp_options
create_egress_only_internet_gateway
create_fleet
create_flow_logs
create_fpga_image
create_image
create_instance_connect_endpoint
create_instance_event_window
create_instance_export_task
create_internet_gateway
create_ipam
create_ipam_pool
create_ipam_resource_discovery
create_ipam_scope
create_key_pair
create_launch_template
create_launch_template_version
create_local_gateway_route
create_local_gateway_route_table
create_local_gateway_route_table_virtual_interface_group_association
create_local_gateway_route_table_vpc_association
create_managed_prefix_list
create_nat_gateway
create_network_acl
create_network_acl_entry
create_network_insights_access_scope
create_network_insights_path
create_network_interface
create_network_interface_permission
create_placement_group
create_public_ipv_4_pool
create_replace_root_volume_task

Cancels the specified Spot Fleet requests
Cancels one or more Spot Instance requests
Determines whether a product code is associated
Copies the specified Amazon FPGA Image (AFI)
Initiates the copy of an AMI
Copies a point-in-time snapshot of an EBS volume
Creates a new Capacity Reservation with the specified
Creates a Capacity Reservation Fleet
Creates a carrier gateway
Creates a Client VPN endpoint
Adds a route to a network to a Client VPN endpoint
Creates a range of customer-owned IP addresses
Creates a pool of customer-owned IP (CoIP) addresses
Provides information to Amazon Web Services about
Creates a default subnet with a size /20 IPv4 CIDR
Creates a default VPC with a size /16 IPv4 CIDR
Creates a custom set of DHCP options
[IPv6 only] Creates an egress-only internet gateway
Creates an EC2 Fleet that contains the configuration
Creates one or more flow logs to capture information
Creates an Amazon FPGA Image (AFI) from the
Creates an Amazon EBS-backed AMI from an AMI
Creates an EC2 Instance Connect Endpoint
Creates an event window in which scheduled events
Exports a running or stopped instance to an Amazon
Creates an internet gateway for use with a VPC
Create an IPAM
Create an IP address pool for Amazon VPC IP Address
Create an IPAM resource discovery
Create an IPAM scope
Creates an ED25519 or 2048-bit RSA key pair with
Creates a launch template
Creates a new version of a launch template
Creates a static route for the specified local gateway
Creates a local gateway route table
Creates a local gateway route table virtual interface
Associates the specified VPC with the specified
Creates a managed prefix list
Creates a NAT gateway in the specified subnet
Creates a network ACL in a VPC
Creates an entry (a rule) in a network ACL with the
Creates a Network Access Scope
Creates a path to analyze for reachability
Creates a network interface in the specified subnet
Grants an Amazon Web Services-authorized account
Creates a placement group in which to launch instances
Creates a public IPv4 address pool
Replaces the EBS-backed root volume for a running
create_reserved_instances_listing
create_restore_image_task
create_route
create_route_table
create_security_group
create_snapshot
create_snapshots
create_spot_datafeed_subscription
create_store_image_task
create_subnet
create_subnet_cidr_reservation
create_tags
create_traffic_mirror_filter
create_traffic_mirror_filter_rule
create_traffic_mirror_session
create_traffic_mirror_target
create_transit_gateway
create_transit_gateway_connect
create_transit_gateway_connect_peer
create_transit_gateway_multicast_domain
create_transit_gateway_peering_attachment
create_transit_gateway_policy_table
create_transit_gateway_prefix_list_reference
create_transit_gateway_route
create_transit_gateway_route_table
create_transit_gateway_route_table_announcement
create_transit_gateway_vpc_attachment
create_verified_access_endpoint
create_verified_access_group
create_verified_access_instance
create_verified_access_trust_provider
create_volume
create_vpc
create_vpc_endpoint
create_vpc_endpoint_connection_notification
create_vpc_endpoint_service_configuration
create_vpc_peering_connection
create_vpn_connection
create_vpn_connection_route
create_vpn_gateway
delete_carrier_gateway
delete_client_vpn_endpoint
delete_client_vpn_route
delete_coip_cidr
delete_coip_pool
delete_customer_gateway
delete_dhcp_options
delete_egress_only_internet_gateway

ec2

Creates a listing for Amazon EC2 Standard Reserved Instances.
Starts a task that restores an AMI from an Amazon S3 object.
Creates a route in a route table within a VPC.
Creates a route table for the specified VPC.
Creates a security group.
Creates a snapshot of an EBS volume and stores it in Amazon S3.
Creates a data feed for Spot Instances, enabling you to store an AMI as a single object in an Amazon S3 bucket.
Creates a subnet in the specified VPC.
Creates a subnet CIDR reservation.
Adds or overwrites only the specified tags for the specified Amazon EC2 resource or resources.
Creates a Traffic Mirror filter.
Creates a Traffic Mirror filter rule.
Creates a Traffic Mirror session.
Creates a target for your Traffic Mirror session.
Creates a transit gateway.
Creates a Connect attachment from a specified transit gateway.
Creates a Connect peer for a specified transit gateway.
Creates a multicast domain using the specified transit gateway.
Requests a transit gateway peering attachment between two VPCs.
Creates a transit gateway policy table.
Creates a reference (route) to a prefix list in a specified transit gateway.
Creates a static route for the specified transit gateway.
Creates a route table for the specified transit gateway.
Advertises a new transit gateway route table.
Attaches the specified VPC to the specified transit gateway.
An Amazon Web Services Verified Access endpoint is where you define your application along with an optional endpoint-level access policy.
An Amazon Web Services Verified Access group is a collection of Amazon Web Services Verified Access endpoints who's associated applications have similar security requirements.
An Amazon Web Services Verified Access instance is a regional entity that evaluates application requests and grants access only when your security requirements are met.
A trust provider is a third-party entity that creates, maintains, and manages identity information for users and devices.
Creates an EBS volume that can be attached to an instance in the same Availability Zone.
Creates a VPC with the specified CIDR blocks.
Creates a VPC endpoint.
Creates a connection notification for a specified VPC.
Creates a VPC endpoint service to which service consumers can connect.
Requests a VPC peering connection between two VPCs.
Creates a VPN connection between an existing VPC and a customer gateway.
Creates a static route associated with a VPN connection.
Deletes a carrier gateway.
Deletes the specified Carrier Gateway endpoint.
Deletes a route from a Carrier Gateway endpoint.
Deletes a range of customer-owned IP addresses.
Deletes a pool of customer-owned IP (CoIP) addresses.
Deletes the specified customer gateway.
Deletes the specified set of DHCP options.
Deletes an egress-only internet gateway.
delete_fleets
delete_flow_logs
delete_fpga_image
delete_instance_connect_endpoint
delete_instance_event_window
delete_internet_gateway
delete_ipam
delete_ipam_pool
delete_ipam_resource_discovery
delete_ipam_scope
delete_key_pair
delete_launch_template
delete_launch_template_versions
delete_local_gateway_route
delete_local_gateway_route_table
delete_local_gateway_route_table_virtual_interface_group_association
delete_local_gateway_route_table_vpc_association
delete_managed_prefix_list
delete_nat_gateway
delete_network_acl
delete_network_acl_entry
delete_network_insights_access_scope
delete_network_insights_access_scope_analysis
delete_network_insights_analysis
delete_network_insights_path
delete_network_interface
delete_network_interface_permission
delete_placement_group
delete_public_ipv_4_pool
delete_queued_reserved_instances
delete_route
delete_route_table
delete_security_group
delete_snapshot
delete_spot_datafeed_subscription
delete_subnet
delete_subnet_cidr_reservation
delete_tags
delete_traffic_mirror_filter
delete_traffic_mirror_filter_rule
delete_traffic_mirror_session
delete_traffic_mirror_target
delete_transit_gateway
delete_transit_gateway_connect
delete_transit_gateway_connect_peer
delete_transit_gateway_multicast_domain
delete_transit_gateway_peering_attachment
delete_transit_gateway_policy_table

Deletes the specified EC2 Fleets
Deletes one or more flow logs
Deletes the specified Amazon FPGA Image (AFI)
Deletes the specified EC2 Instance Connect Endpoint
Deletes the specified event window
Deletes the specified internet gateway
Delete an IPAM
Delete an IPAM pool
Deletes an IPAM resource discovery
Delete the scope for an IPAM
Deletes the specified key pair, by removing the public key from Amazon EC2
Deletes a launch template
Deletes one or more versions of a launch template
Deletes the specified route from the specified local gateway route table
Deletes a local gateway route table
Deletes a local gateway route table virtual interface group association
Deletes the specified association between a VPC and a local gateway
Deletes the specified managed prefix list
Deletes the specified NAT gateway
Deletes the specified network ACL
Deletes the specified ingress or egress entry (rule)
Deletes the specified Network Access Scope
Deletes the specified Network Access Scope analysis
Deletes the specified network insights analysis
Deletes the specified path
Deletes the specified network interface
Deletes a permission for a network interface
Deletes the specified placement group
Delete a public IPv4 pool
Deletes the queued purchases for the specified Reserved Instances
Deletes the specified route from the specified route table
Deletes the specified route table
Deletes a security group
Deletes the specified snapshot
Deletes the data feed for Spot Instances
Deletes the specified subnet
Deletes a subnet CIDR reservation
Deletes the specified set of tags from the specified resource
Deletes the specified Traffic Mirror filter
Deletes the specified Traffic Mirror rule
Deletes the specified Traffic Mirror session
Deletes the specified Traffic Mirror target
Deletes the specified transit gateway
Deletes the specified Connect attachment
Deletes the specified Connect peer
Deletes the specified transit gateway multicast domain
Deletes a transit gateway peering attachment
Deletes the specified transit gateway policy table
delete_transit_gateway_prefix_list_reference
delete_transit_gateway_route
delete_transit_gateway_route_table
delete_transit_gateway_route_table_announcement
delete_transit_gateway_vpc_attachment
delete_verified_access_endpoint
delete_verified_access_group
delete_verified_access_instance
delete_verified_access_trust_provider
delete_volume
delete_vpc
delete_vpc_endpoint_connection_notifications
delete_vpc_endpoints
delete_vpc_peering_connection
delete_vpn_connection
delete_vpn_connection_route
delete_vpn_gateway
deprovision_byoip_cidr
deprovision_ipam_byoasn
deprovision_ipam_pool_cidr
deprovision_public_ipv4_pool_cidr
deregister_image
deregister_instance_event_notification_attributes
deregister_transit_gateway_multicast_group_members
deregister_transit_gateway_multicast_group_sources
describe_account_attributes
describe_addresses
describe_addresses_attribute
describe_address_transfers
describe_aggregate_id_format
describe_availability_zones
describe.aws.network.performance.metric.subscriptions
describe_bundle_tasks
describe_byoip_cidrs
describe_capacity_block_offerings
describe_capacity_reservation_fleets
describe_capacity_reservations
describe_carrier_gateways
describe_classic_link_instances
describe_client_vpn_authorization_rules
describe_client_vpn_connections
describe_client_vpn_endpoints
describe_client_vpn_routes
describe_client_vpn_target_networks
describe_coip_pools
describe_conversion_tasks
describe_customer_gateways

deletes a reference (route) to a prefix list in a specified transit gateway route table.
deletes the specified route from the specified transit gateway route table.
Advertises to the transit gateway that a transit gateway route table
is being deleted.
deletes the specified VPC attachment.
delete an Amazon Web Services Verified Access endpoint.
delete an Amazon Web Services Verified Access group.
delete an Amazon Web Services Verified Access instance.
delete an Amazon Web Services Verified Access trust provider.
deletes the specified EBS volume.
deletes the specified VPC.
deletes the specified VPC endpoint connection notification.
deletes the specified VPC endpoints.
deletes the specified VPC endpoint service configurations.
deletes a VPC peering connection.
deletes the specified VPN connection.
deletes the specified static route associated with a VPC endpoint service configuration.
deletes the specified virtual private gateway.
releases the specified address range that you provisioned for use with your Amazon Web Services resources through bring your own IP addresses (BYOIP) and deletes the corresponding address pool.
deprovisions your Autonomous System Number (ASN) from your Amazon Web Services account.
deprovision a CIDR provisioned from an IPAM.
deprovision a CIDR from a public IPv4 pool.
deregisters the specified AMI.
deregisters tag keys to prevent tags that have the specified tag keys from being included in scheduled event notifications for resources in the Region.
deregisters the specified sources (network interfaces) from a transit gateway multicast group.
deregisters the specified members (network interfaces) from a transit gateway multicast group.
describes attributes of your Amazon Web Services account.
describes the attributes of the specified Elastic IP addresses or all Elastic IP addresses.
describes the Availability Zones, Local Zones, and Wavelength Zones that are available to you.
describes attributes of your Amazon Web Services account.
describes the current Infrastructure Performance metric subscriptions.
describes the specified bundle tasks or all of your bundle tasks.
describes the IP address ranges that were specified when you provisioned for use with your Amazon Web Services resources through bring your own IP addresses (BYOIP) and deletes the corresponding address pool.
describes the Capacity Block offerings available for purchase in the Amazon Web Services Region that you're currently using.
describes one or more Capacity Reservation Fleets.
describes one or more of your Capacity Reservations.
describes one or more of your carrier gateways.
This action is deprecated.
describes the authorization rules for a specified Client VPN endpoint.
describes active client connections and connection requests.
describes one or more Client VPN endpoints in the specified Region.
describes the routes for the specified Client VPN endpoint.
describes the target networks associated with the specified route.
describes the specified customer-owned address pool.
describes the specified conversion tasks or all of your conversion tasks.
describe_dhcp_options
describe_egress_only_internet_gateways
describe_elastic_gpus
describe_export_image_tasks
describe_export_tasks
describe_fast_launch_images
describe_fast_snapshot_restores
describe_fleet_history
describe_fleet_instances
describe_fleets
describe_flow_logs
describe_fpga_image_attribute
describe_fpga_images
describe_host_reservation_offerings
describe_host_reservations
describe_hosts
describe_iam_instance_profile_associations
describe_identity_id_format
describe_id_format
describe_image_attribute
describe_images
describe_import_image_tasks
describe_import_snapshot_tasks
describe_instance_attribute
describe_instance_connect_endpoints
describe_instance_credit_specifications
describe_instance_event_notification_attributes
describe_instance_event_windows
describe_instances
describe_instance_status
describe_instance_topology
describe_instance_type_offerings
describe_instance_types
describe_internet_gateways
describe_ipam_byoasn
describe_ipam_pools
describe_ipam_resource_discoveries
describe_ipam_resource_discovery_associations
describe_ips
describe_ipam_scopes
describe_ipv_6_pools
describe_key_pairs
describe_launch_templates
describe_launch_template_versions
describe_local_gateway_route_tables
describe_local_gateway_route_table_virtual_interface_group_associations
describe_local_gateway_route_table_vpc_associations
describe_local_gateways

Describes one or more of your DHCP options sets.
Describes one or more of your egress-only internet gateways.
Amazon Elastic Graphics reached end of life on January 8, 2024.
Describes the state of fast snapshot restores for your AMIs.
Describes the running instances for the specified EC2 Fleet or all of your EC2 Fleets.
Describes the specified EC2 Fleet or all of your EC2 Fleets.
Describes one or more flow logs.
Describes the specified attribute of the specified AMI.
Describes the Amazon FPGA Images (AFIs) available to you.
Describes the Dedicated Host reservations that are or were associated with Dedicated Hosts.
Describes your IAM instance profile associations.
Describes the ID format settings for resources for the specified IAM user, IAM role, or root user.
Describes the ID format settings for your resources on a per-Region basis.
Describes the AMIs, AKIs, and ARIs available to you.
Displays details about an import virtual machine or import snapshot tasks that are already created.
Describes your import snapshot tasks.
Describes the specified attribute of the specified instance.
Describes the specified event windows or all event windows.
Describes the specified instances or all instances.
Describes the status of the specified instances or all instances.
Describes a tree-based hierarchy that represents the physical host placement of your EC2 instances within an Availability Zone.
Lists the instance types that are offered for the specified location.
Describes one or more of your internet gateways.
Describes your Autonomous System Numbers (ASNs).
Get information about your IPAM pools.
Describes IPAM resource discoveries.
Describes resource discovery associations with an IPAM pool.
Get information about your IPAM pools.
Get information about your IPAM scopes.
Describes your IPv6 address pools.
Describes the specified key pairs or all of your key pairs.
Describes one or more launch templates.
Describes one or more versions of a specified launch template.
Describes one or more local gateway route tables.
Describes the associations between virtual interface groups and local gateway route tables.
Describes the specified associations between VPCs and local gateway route tables.
Describes one or more local gateways.
describe_local_gateway_virtual_interface_groups
Describes the specified local gateway virtual interface groups

describe_local_gateway_virtual_interfaces
Describes the specified local gateway virtual interfaces

describe_locked_snapshots
Describes the lock status for a snapshot

describe_mac_hosts
Describes the specified EC2 Mac Dedicated Hosts

describe_managed_prefix_lists
Describes your managed prefix lists and any Amazon Web Services-managed prefix lists

This action is deprecated

describe_nat_gateways
Describes one or more of your NAT gateways

describe_network_acls
Describes one or more of your network ACLs

describe_network_insights_access_scope_analyses
Describes the specified Network Access Scope analyses

describe_network_insights_access_scopes
Describes the specified Network Access Scopes

Describes one or more of your network insights analyses

describe_network_insights_analyses
Describes one or more of your network insights analyses

describe_network_insights_paths
Describes one or more of your paths

describe_network_interface_attribute
Describes a network interface attribute

describe_network_interface_permissions
Describes the permissions for your network interfaces

describe_network_interfaces
Describes one or more of your network interfaces

describe_placement_groups
Describes the specified placement groups or all of your placement groups

describe_prefix_lists
Describes available Amazon Web Services services in a prefix list format, which includes the prefix list name and prefix list ID of the service and the IP address range for the service

describe_principal_id_format
Describes the ID format settings for the root user and all IAM roles and IAM users that have explicitly specified a longer ID (17-character ID) preference

describe_public_ipv_4_pools
Describes the specified IPv4 address pools

describe_regions
Describes the Regions that are enabled for your account, or all Regions

describe_replace_root_volume_tasks
Describes a root volume replacement task

describe_reserved_instances
Describes one or more of the Reserved Instances that you purchased

describe_reserved_instances_listings
Describes your account’s Reserved Instance listings

describe_reserved_instances_modifications
Describes the modifications made to your Reserved Instances

describe_reserved_instances_offerings
Describes Reserved Instance offerings that are available in the Reserved Instance Marketplace

Describes one or more of your route tables

describe_route_tables
Finds available schedules that meet the specified criteria

describe_scheduled_instance_availability
Describes the specified Scheduled Instances or all of your scheduled instances

describe_security_group_references
Describes the VPCs on the other side of a VPC peering connection that are referencing the security groups you've specified in this request

describe_security_group_rules
Describes one or more of your security group rules

describe_security_groups
Describes the specified security groups or all of your security groups

Describes the specified attribute of the specified security group

describe_snapshot_attribute
Describes the specified attribute of the specified snapshot

describe_snapshots
Describes the specified EBS snapshots available to you or all of the EBS snapshots available to you

Describes the storage tier status of one or more Amazon EBS snapshots

describe_snapshot_tier_status
Describes the data feed for Spot Instances

describe_spot_datafeed_subscription
Describes the running instances for the specified AMI

describe_spot_fleet_instances
Describes your Spot Fleet requests

describe_spot_fleet_request_history
Describes the events for the specified Spot Fleet request

describe_spot_fleet_requests
Describes your Spot Fleet requests

Describes the specified Spot Instance requests

describe_spot_instance_requests
Describes the Spot price history

describe_stale_security_groups
Describes the stale security group rules for security groups in a specified VPC

describe_store_image_tasks
Describes the progress of the AMI store tasks

describe_subnets
Describes one or more of your subnets

describe_tags
Describes the specified tags for your EC2 resources

Describes one or more Traffic Mirror filters

describe_traffic_mirror_filters
Describes one or more Traffic Mirror sessions

describe_traffic_mirror_sessions
Describes one or more Traffic Mirror targets

describe_traffic_mirror_targets
Describes one or more attachments between resources and transit gateways

describe_transit_gateway_attachments
describe_transit_gateway_connect_peers Describes one or more Connect peers
describe_transit_gateway_connects Describes one or more Connect attachments
describe_transit_gateway_multicast_domains Describes one or more transit gateway multicast domains
describe_transit_gateway_peering_attachments Describes your transit gateway peering attachments
describe_transit_gateway_policy_tables Describes one or more transit gateway route policy tables
describe_transit_gateway_route_table_announcements Describes one or more transit gateway route table announcements
describe_transit_gateway_route_tables Describes one or more transit gateway route tables
describe_transit_gateways Describes one or more transit gateways
describe_transit_gateway_vpc_attachments Describes one or more VPC attachments
describe_trunk_interface_associations Describes one or more network interface trunk associations
describe_verified_access_endpoints Describes the specified Amazon Web Services Verified Access endpoints
describe_verified_access_groups Describes the specified Verified Access groups
describe_verified_access_instance_logging_configurations Describes the specified Amazon Web Services Verified Access instances
describe_verified_access_instance_permissions Describes the specified attribute of the specified Verified Access instance
describe_verified_access_instances Describes the specified Amazon Web Services Verified Access instances
describe_verified_access_trust_providers Describes the specified Amazon Web Services Verified Access trust providers
describe_volume_attribute Describes the specified attribute of the specified volume
describe_volumes Describes the specified EBS volumes or all of your EBS volumes
describe_volumes_modifications Describes the most recent volume modification request for the specified EBS volumes
describe_volume_status Describes the status of the specified volumes
describe_vpc_attribute Describes the specified attribute of the specified VPC
describe_vpc_classic_link This action is deprecated
describe_vpc_classic_link_dns_support This action is deprecated
describe_vpc_endpoint_connection_notifications Describes the connection notifications for VPC endpoint services
describe_vpc_endpoint_connections Describes the VPC endpoint connections to your VPC endpoints
describe_vpc_endpoint_service_configurations Describes the VPC endpoint service configurations in your account (your services)
describe_vpc_endpoint_service_permissions Describes the principals (service consumers) that can use your VPC endpoint services
describe_vpc_endpoint_services Describes available services to which you can create a VPC endpoint
describe_vpc_peering_connections Describes one or more of your VPC peering connections
describe_vpc_peering_attachments Describes one or more of your VPC peering attachments
describe_vpcs Describes one or more of your VPCs
detach_classic_link_vpc This action is deprecated
detach_internet_gateway Detaches an internet gateway from a VPC, disabling connectivity between the internet and the VPC
detach_network_interface Detaches a network interface from an instance
detach_verified_access_trust_provider Detaches the specified Amazon Web Services Verified Access trust provider from the specified Amazon Web Services Verified Access instance
detach_volume Detaches an EBS volume from an instance
detach_vpn_gateway Detaches a virtual private gateway from a VPC
disable_address_transfer Disables Elastic IP address transfer
disable_aws_network_performance_metric_subscription Disables Infrastructure Performance metric subscriptions
disable_ebs_encryption_by_default Disables EBS encryption by default for your account
disable_fast_launch Discontinue Windows fast launch for a Windows AMI, and clean up existing pre-provisioned snapshots
disable_fast_snapshot_restores Disables fast snapshot restores for the specified snapshots in the specified Availability Zones
disable_image Sets the AMI state to disabled and removes all launch permissions from the AMI
disable_image_block_public_access Disables block public access for AMIs at the account level in the specified Amazon Web Services Region
disable_image_deprecation Cancels the deprecation of the specified AMI
disable_image_deregistration_protection Disables deregistration protection for an AMI
disable_ipam_organization_admin_account Disable the IPAM account
 disables access to the EC2 serial console of all instances.
Disables block public access for snapshots set to private.
Disables the specified resource attachment from propagating routes.
Disables a virtual private gateway (VGW) from propagating routes.
This action is deprecated.
This action is deprecated.
Disassociates an Elastic IP address from the instance.
Disassociates a target network from the specified VPC.
Disassociates an IAM role from an IAM instance profile.
Disassociates an IAM instance profile from a resource.
Removes the association between a VPC and a NAT gateway.
Disassociates a subnet or gateway from a VPC.
Disassociates a CIDR block from a subnet.
Disassociates the specified subnets from the transit gateway.
Removes the association between an attachment and a policy table.
Removes an association between a branch network interface and a trunk network interface.
Disassociates a CIDR block from a VPC.
Enables Elastic IP address transfer.
Enables Infrastructure Performance subscriptions.
Enables EBS encryption by default for your account.
When enabled, Windows fast launch is available.
Enables fast snapshot restores for the specified snapshot.
Re-enables a disabled AMI.
Enables block public access for AMIs at the account level.
Enables deregistration protection for an AMI.
Enable an Organizations member account as the IPAM admin account.
Establishes a trust relationship between Reachability Analyzer and Organizations.
Enables access to the EC2 serial console of all instances.
Enables or modifies the block public access for snapshots.
Enables the specified attachment to propagate routes.
Enables a virtual private gateway (VGW) to propagate routes.
Enables I/O operations for a volume that had I/O operations disabled because the data on the volume was potentially inconsistent.
This action is deprecated.
This action is deprecated.
Downloads the client certificate revocation list for the specified Account ID.
Downloads the contents of the Client VPN endpoint configuration file for the specified Amazon Machine Image (AMI) to a VPC.
Exports routes from the specified transit gateway.
Retrieves the IAM roles that are associated with the specified account.
Gets information about the IPv6 CIDR block associations for an address pool.
Gets network performance data.
Gets usage information about a Capacity Reservation.
import_snapshot
import_key_pair
import_instance
import_image
import_client_vpn_client_certificate_revocation_list
get_vpn_tunnel_replacement_status
get_vpn_connection_device_types
get_vpn_connection_device_sample_configuration
get_verified_access_group_policy
get_verified_access_endpoint_policy
get_transit_gateway_route_table_propagations
get_transit_gateway_prefix_list_references
get_transit_gateway_policy_table_entries
get_transit_gateway_policy_table_associations
get_transit_gateway_multicast_domain Associations
get_transit_gateway_attachment_propagations
get_subnet_cidr_reservations
get_spot_placement_scores
get_snapshot_block_public_access_state
get_serial_console_access_status
get_security_groups_for_vpc
get_reserved_instances_exchange_quote
get_image_block_public_access_state
get_instance_types_from_instance_requirements
get_instance_metadata_defaults
get_instance_uefi_data
get_ipam_address_history
get_ipam_discovered_accounts
get_ipam_discovered_public_addresses
get_ipam_discovered_resource_cidrs
get_ipam_pool_allocations
get_ipam_pool_cids
get_ipam_resource_cids
get_launch_template_data
get_managed_prefix_list_associations
get_managed_prefix_list_entries
get_network_insights_access_scope_analysis_findings
get_network_insights_access_scope_content
get_password_data
get_groups_for_capacity_reservation
get_flow_logs_integration_template
get_host_reservation_purchase_preview
get_ebs_encryption_by_default
get_ebs_default_kms_key_id
get_default_credit_specification
get_console_screenshot
get_default_console_access_status
get_ipam_address_history
get_ipam_discovery_accounts
get_ipam_discovered_public_addresses
get_ipam_discovered_resource_cidrs
get_ipam_pool_allocations
get_ipam_pool_cids
get_ipam_resource_cids
get_launch_template_data
get_managed_prefix_list_associations
get_launch_template_data
get_network_insights_access_scope_analysis_findings
get_network_insights_access_scope_content
get_password_data
get_groups_for_capacity_reservation
get_flow_logs_integration_template
get_host_reservation_purchase_preview
get_ebs_encryption_by_default
get_ebs_default_kms_key_id
get_default_credit_specification
get_console_screenshot
get_default_console_access_status
get_ipam_address_history
get_ipam_discovery_accounts
get_ipam_discovered_public_addresses
get_ipam_discovered_resource_cidrs
get_ipam_pool_allocations
get_ipam_pool_cids
get_ipam_resource_cids
get_launch_template_data
get_managed_prefix_list_associations
get_launch_template_data
get_network_insights_access_scope_analysis_findings
get_network_insights_access_scope_content
get_password_data
get_groups_for_capacity_reservation
get_flow_logs_integration_template
get_host_reservation_purchase_preview
get_ebs_encryption_by_default
get_ebs_default_kms_key_id
get_default_credit_specification
get_console_screenshot
get_default_console_access_status
get_ipam_address_history
get_ipam_discovery_accounts
get_ipam_discovered_public_addresses
get_ipam_discovered_resource_cidrs
get_ipam_pool_allocations
get_ipam_pool_cids
get_ipam_resource_cids
get_launch_template_data
get_managed_prefix_list_associations
get_launch_template_data
get_network_insights_access_scope_analysis_findings
get_network_insights_access_scope_content
get_password_data
get_groups_for_capacity_reservation
get_flow_logs_integration_template
get_host_reservation_purchase_preview
get_ebs_encryption_by_default
get_ebs_default_kms_key_id
get_default_credit_specification
get_console_screenshot
get_default_console_access_status
get_ipam_address_history
get_ipam_discovery_accounts
get_ipam_discovered_public_addresses
get_ipam_discovered_resource_cidrs
get_ipam_pool_allocations
get_ipam_pool_cids
get_ipam_resource_cids
get_launch_template_data
get_managed_prefix_list_associations
get_launch_template_data
get_network_insights_access_scope_analysis_findings
get_network_insights_access_scope_content
get_password_data
get_groups_for_capacity_reservation
get_flow_logs_integration_template
get_host_reservation_purchase_preview
get_ebs_encryption_by_default
get_ebs_default_kms_key_id
get_default_credit_specification
get_console_screenshot
get_default_console_access_status
get_ipam_address_history
get_ipam_discovery_accounts
get_ipam_discovered_public_addresses
get_ipam_discovered_resource_cidrs
get_ipam_pool_allocations
get_ipam_pool_cids
get_ipam_resource_cids
get_launch_template_data
get_managed_prefix_list_associations
get_launch_template_data
get_network_insights_access_scope_analysis_findings
get_network_insights_access_scope_content
get_password_data
get_groups_for_capacity_reservation
get_flow_logs_integration_template
get_host_reservation_purchase_preview
get_ebs_encryption_by_default
get_ebs_default_kms_key_id
get_default_credit_specification
get_console_screenshot
get_default_console_access_status
get_ipam_address_history
get_ipam_discovery_accounts
get_ipam_discovered_public_addresses
get_ipam_discovered_resource_cidrs
get_ipam_pool_allocations
get_ipam_pool_cids
get_ipam_resource_cids
get_launch_template_data
get_managed_prefix_list_associations
get_launch_template_data
get_network_insights_access_scope_analysis_findings
get_network_insights_access_scope_content
get_password_data
get_groups_for_capacity_reservation
get_flow_logs_integration_template
get_host_reservation_purchase_preview
get_ebs_encryption_by_default
get_ebs_default_kms_key_id
get_default_credit_specification
get_console_screenshot
get_default_console_access_status
get_ipam_address_history
get_ipam_discovery_accounts
get_ipam_discovered_public_addresses
get_ipam_discovered_resource_cidrs
get_ipam_pool_allocations
get_ipam_pool_cids
get_ipam_resource_cids
get_launch_template_data
get_managed_prefix_list_associations
get_launch_template_data
get_network_insights_access_scope_analysis_findings
get_network_insights_access_scope_content
get_password_data
get_groups_for_capacity_reservation
get_flow_logs_integration_template
get_host_reservation_purchase_preview
get_ebs_encryption_by_default
get_ebs_default_kms_key_id
get_default_credit_specification
get_console_screenshot
get_default_console_access_status
get_ipam_address_history
get_ipam_discovery_accounts
get_ipam_discovered_public_addresses
get_ipam_discovered_resource_cidrs
get_ipam_pool_allocations
get_ipam_pool_cids
get_ipam_resource_cids
get_launch_template_data
get_managed_prefix_list_associations
get_launch_template_data
get_network_insights_access_scope_analysis_findings
get_network_insights_access_scope_content
get_password_data
get_groups_for_capacity_reservation
get_flow_logs_integration_template
get_host_reservation_purchase_preview
get_ebs_encryption_by_default
get_ebs_default_kms_key_id
get_default_credit_specification
get_console_screenshot
get_default_console_access_status
get_ipam_address_history
get_ipam_discovery_accounts
get_ipam_discovered_public_addresses
get_ipam_discovered_resource_cidrs
get_ipam_pool_allocations
get_ipam_pool_cids
get_ipam_resource_cids
get_launch_template_data
get_managed_prefix_list_associations
get_launch_template_data
get_network_insights_access_scope_analysis_findings
get_network_insights_access_scope_content
get_password_data
get_groups_for_capacity_reservation
get_flow_logs_integration_template
get_host_reservation_purchase_preview
get_ebs_encryption_by_default
get_ebs_default_kms_key_id
get_default_credit_specification
get_console_screenshot
get_default_console_access_status
get_ipam_address_history
get_ipam_discovery_accounts
get_ipam_discovered_public_addresses
get_ipam_discovered_resource_cidrs
get_ipam_pool_allocations
get_ipam_pool_cids
get_ipam_resource_cids
get_launch_template_data
get_managed_prefix_list_associations
get_launch_template_data
get_network_insights_access_scope_analysis_findings
get_network_insights_access_scope_content
get_password_data
get_groups_for_capacity_reservation
get_flow_logs_integration_template
get_host_reservation_purchase_preview
get_ebs_encryption_by_default
get_ebs_default_kms_key_id
get_default_credit_specification
get_console_screenshot
get_default_console_access_status
get_ipam_address_history
get_ipam_discovery_accounts
get_ipam_discovered_public_addresses
get_ipam_discovered_resource_cidrs
get_ipam_pool_allocations
get_ipam_pool_cids
get_ipam_resource_cids
get_launch_template_data
get_managed_prefix_list_associations
get_launch_template_data
get_network_insights_access_scope_analysis_findings
get_network_insights_access_scope_content
get_password_data
get_groups_for_capacity_reservation
get_flow_logs_integration_template
get_host_reservation_purchase_preview
get_ebs_encryption_by_default
get_ebs_default_kms_key_id
get_default_credit_specification
get_console_screenshot
get_default_console_access_status
get_ipam_address_history
get_ipam_discovery_accounts
get_ipam_discovered_public_addresses
get_ipam_discovered_resource_cidrs
get_ipam_pool_allocations
get_ipam_pool_cids
get_ipam_resource_cids
get_launch_template_data
get_managed_prefix_list_associations
get_launch_template_data
get_network_insights_access_scope_analysis_findings
get_network_insights_access_scope_content
get_password_data
get_groups_for_capacity_reservation
get_flow_logs_integration_template
get_host_reservation_purchase_preview
get_ebs_encryption_by_default
get_ebs_default_kms_key_id
get_default_credit_specification
get_console_screenshot
get_default_console_access_status
get_ipam_address_history
get_ipam_discovery_accounts
get_ipam_discovered_public_addresses
get_ipam_discovered_resource_cidrs
gets the console output for the specified instance
Retrieve a JPG-format screenshot of a running instance
Describes the default credit option for CPU usage
Describes the default KMS key for EBS encryption
Generates a CloudFormation template that streamlines and automates the integration of VPC flow logs with Amazon Athena
Gets the current state of block public access for AMI
Gets the default instance metadata service (IMDS)
Returns a list of instance types with the specified criteria
A binary representation of the UEFI variable store
Retrieve historical information about a CIDR with IPAM
Gets IPAM discovered accounts
Gets the public IP addresses that have been discovered
Returns the list of all CIDR allocations in an IPAM pool
Get the CIDRs provisioned to an IPAM pool
Returns resource CIDRs managed by IPAM in a scope
Retrieves the configuration data of the specified instance
Gets information about the resources that are associated with a CIDR
Gets the findings for the specified Network Access Scope
Gets the content for the specified Network Access Scope
Retrieves the encrypted administrator password for a specified instance
Returns a quote and exchange information for exchange
Gets security groups that can be associated by the current AWS account
Retrieves the access status of your account to the specified CIDR
Gets the current state of block public access for subnets
Calculates the Spot placement score for a Region
Gets information about the subnet CIDR reservations
Lists the route tables to which the specified resource is associated
Gets information about the associations for the transit gateway policy table
Returns a list of transit gateway policy table entries
Gets information about the prefix list references in the transit gateway policy
Gets information about the associations for the specified prefix list reference
Gets information about the route table propagation
Get the Verified Access policy associated with the resource
Shows the contents of the Verified Access policy
Download an Amazon Web Services-provided sample configuration file
Obtain a list of customer gateway devices for which a Site-to-Site VPN connection is available
Get details of available tunnel endpoint maintenance status
To import your virtual machines (VMs) with a console-based experience, you can use the Import virtual machine images to Amazon Web Services template in the Migration Hub Orchestrator console
Imports the public key from an RSA or ED25519 key pair
Imports a disk into an EBS snapshot
import_volume
list_images_in_recycle_bin
list_snapshots_in_recycle_bin
lock_snapshot
modify_address_attribute
modify_availability_zone_group
modify_capacity_reservation
modify_capacity_reservation_fleet
modify_client_vpn_endpoint
modify_default_credit_specification
modify_ebs_default_kms_key_id
modify_fleet
modify_fpga_image_attribute
modify_hosts
modify_identity_id_format
modify_id_format
modify_image_attribute
modify_instance_attribute
modify_instance_capacity_reservation_attributes
modify_instance_credit_specification
modify_instance_event_start_time
modify_instance_event_window
modify_instance_maintenance_options
modify_instance_metadata_defaults
modify_instance_metadata_options
modify_instance_placement
modify_ipam
modify_ipam_pool
modify_ipam_resource_cidr
modify_ipam_resource_discovery
modify_ipam_scope
modify_launch_template
modify_local_gateway_route
modify_managed_prefix_list
modify_network_interface_attribute
modify_private_dns_name_options
modify_reserved_instances
modify_security_group_rules
modify_snapshot_attribute
modify_snapshot_tier
modify_spot_fleet_request
modify_subnet_attribute
modify_traffic_mirror_filter_network_services
modify_traffic_mirror_filter_rule
modify_traffic_mirror_session
modify_transit_gateway
modify_transit_gateway_prefix_list_reference
modify_transit_gateway_vpc_attachment

ec2

Creates an import volume task using metadata from the specified disk image.
Lists one or more AMIs that are currently in the Recycle Bin.
Lists one or more snapshots that are currently in the Recycle Bin.
Locks an Amazon EBS snapshot in either governance or compliance mode to protect it against accidental or malicious deletions for a specific duration.
Modifies an attribute of the specified Elastic IP address.
Modifies the opt-in status of the Local Zone and Wavelength Zone group for your account.
Modifies a Capacity Reservation’s capacity and the conditions under which it is to be released.
Modifies a Capacity Reservation Fleet.
Modifies the specified Client VPN endpoint.
Modifies the default credit option for CPU usage on a running or stopped burstable performance instance.
Modifies the specified EC2 Fleet.
Modifies the specified attribute of the specified AMI.
Modifies the auto-placement setting of a Dedicated Host.
Modifies the ID format of a resource for a specified AWS service.
Modifies the ID format for the specified resource.
Modifies the specified attribute of the specified AMI.
Modifies the specified attribute of the specified instance.
Modifies the Capacity Reservation settings for a running or stopped instance.
Modifies the credit option for CPU usage on a running or stopped burstable performance instance.
Modifies the start time for a scheduled Amazon EC2 instance event.
Modifies the specified event window.
Modifies the recovery behavior of your instance to disable simplified automatic recovery or set the recovery behavior to default.
Modifies the default instance metadata service (IMDS) settings at the account level in the specified Amazon Web Services Region.
Modifies the instance metadata parameters on a running or stopped instance.
Modifies the placement attributes for a specified instance.
Modifies the configurations of an IPAM.
Modifies the configurations of an IPAM pool.
Modifies a resource CIDR.
Modifies a resource discovery.
Modifies an IPAM scope.
Modifies a launch template.
Modifies the specified local gateway route.
Modifies the specified managed prefix list.
Modifies the specified network interface attribute.
Modifies the options for instance hostnames for the specified instance family.
Modifies the configuration of your Reserved Instances.
Modifies the rules of a security group.
Adds or removes permission settings for the specified Amazon EBS snapshot.
Modifies the specified Spot Fleet request.
Modifies a subnet attribute.
Allows or restricts mirroring network services.
Modifies the specified Traffic Mirror rule.
Modifies a Traffic Mirror session.
Modifies the specified transit gateway.
Modifies a reference (route) to a prefix list in a specified VPC.
Modifies the specified VPC attachment.
modify_verified_access_endpoint
modify_verified_access_endpoint_policy
modify_verified_access_group
modify_verified_access_group_policy
modify_verified_access_instance
modify_verified_access_instance_logging_configuration
modify_verified_access_trust_provider
modify_volume
modify_volume_attribute
modify_vpc_attribute
modify_vpc_endpoint
modify_vpc_endpoint_connection_notification
modify_vpc_endpoint_service_configuration
modify_vpc_endpoint_service_payer_responsibility
modify_vpc_endpoint_service_permissions
modify_vpc_peering_connection_options
modify_vpc_tenancy
modify_vpn_connection
modify_vpn_connection_options
modify_vpn_tunnel_certificate
modify_vpn_tunnel_options
monitor_instances
move_address_to_vpc
move_byoip_cidr_to_ipam
provision_byoip_cidr
provision_ipam_byoasn
provision_ipam_pool_cidr
provision_public_ipv4_pool_cidr
purchase_capacity_block
purchase_host_reservation
purchase_reserved_instances_offering
purchase_scheduled_instances
reboot_instances
register_image
register_instance_event_notification_attributes
register_transit_gateway_multicast_group_members
register_transit_gateway_multicast_group_sources
reject_transit_gateway_multicast_domain_associations
reject_transit_gateway_peering_attachment
reject_transit_gateway_vpc_attachment
reject_vpc_endpoint_connections
reject_vpc_peering_connection
release_address
release_hosts
release_ipam_pool_allocation
replace_iam_instance_profile_association
replace_network_acl_association
replace_network_acl_entry

Modifies the configuration of the specified Amazon Web Services Verified Access endpoint
Modifies the specified Amazon Web Services Verified Access endpoint policy
Modifies the configuration of the specified Amazon Web Services Verified Access group
Modifies the specified Amazon Web Services Verified Access group policy
Modifies the configuration of the specified Amazon Web Services Verified Access instance
Modifies the logging configuration for the specified Amazon Web Services Verified Access instance
Modifies the configuration of the specified Amazon Web Services Verified Access trust provider
You can modify several parameters of an existing EBS volume
Modifies a volume attribute
Modifies the specified attribute of the specified EBS volume
Modifies attributes of a specified VPC endpoint
Modifies a connection notification for VPC endpoint
Modifies the attributes of your VPC endpoint service
Modifies the payer responsibility for your VPC endpoint service
Modifies the permissions for your VPC endpoint service
Modifies the VPC peering connection options on a specified VPC endpoint
Modifies the instance tenancy attribute of the specified VPC
Modifies the customer gateway or the target gateway
Modifies the connection options for your Site-to-Site VPN connection
Modifies the VPN tunnel endpoint certificate
Modifies the options for a VPN tunnel in an Amazon VPC
Enables detailed monitoring for a running instance
This action is deprecated
Move a BYOIPv4 CIDR to IPAM from a public IPv4 pool
Provisions an IPv4 or IPv6 address range for use with your Autonomous System Number (ASN)
Provision a CIDR to an IPAM pool
Provision a CIDR to a public IPv4 pool
Purchase the Capacity Block for use with your account
Purchase a reservation with configurations that match those of your Dedicated Host
Purchases a Reserved Instance for use with your account
You can no longer purchase Scheduled Instances
Requests a reboot of the specified instances
Registers an AMI
Registers a set of tag keys to include in scheduled event notifications
Registers members (network interfaces) with the transit gateway
Registers sources (network interfaces) with the specified transit gateway
Rejects a request to associate cross-account subnet
Rejects a transit gateway peering attachment request
Rejects a request to attach a VPC to a transit gateway
Rejects VPC endpoint connection requests to your VPC
Releases the specified Elastic IP address
When you no longer want to use an On-Demand Instance
Release an allocation within an IPAM pool
Replaces an IAM instance profile for the specified network
Changes which network ACL a subnet is associated with
Replaces an entry (rule) in a network ACL
replace_route
replace_route_table_association
replace_transit_gateway_route
replace_vpn_tunnel
report_instance_status
request_spot_fleet
request_spot_instances
reset_address_attribute
reset_ebs_default_kms_key_id
reset_fpga_image_attribute
reset_image_attribute
reset_instance_attribute
reset_network_interface_attribute
reset_snapshot_attribute
restore_address_to_classic
restore_image_from_recycle_bin
restore_managed_prefix_list_version
restore_snapshot_from_recycle_bin
restore_snapshot_tier
revoke_client_vpn_ingress
revoke_security_group_egress
revoke_security_group_ingress
run_instances
run_scheduled_instances
search_local_gateway_routes
search_transit_gateway_multicast_groups
search_transit_gateway_routes
send_diagnostic_interrupt
start_instances
start_network_insights_access_scope_analysis
start_network_insights_analysis
start_vpc_endpoint_service_private_dns_verification
stop_instances
terminate_client_vpn_connections
terminate_instances
unassign_ipv_6_addresses
unassign_private_ip_addresses
unassign_private_nat_gateway_address
unlock_snapshot
unmonitor_instances
update_security_group_rule_descriptions_egress
update_security_group_rule_descriptions_ingress
withdraw_byoip_cidr

Examples

## Not run:
ec2instanceconnect <- ec2()
# This example allocates an Elastic IP address.
svc$allocate_address()

## End(Not run)

### Description

This is the *Amazon EC2 Instance Connect API Reference*. It provides descriptions, syntax, and usage examples for each of the actions for Amazon EC2 Instance Connect. Amazon EC2 Instance Connect enables system administrators to publish one-time use SSH public keys to EC2, providing users a simple and secure way to connect to their instances.

To view the Amazon EC2 Instance Connect content in the *Amazon EC2 User Guide*, see Connect to your Linux instance using EC2 Instance Connect.

For Amazon EC2 APIs, see the *Amazon EC2 API Reference*.

### Usage

```r
ec2instanceconnect(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

### Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• `s3_force_path_style`: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.

• `sts_regional_endpoint`: Set sts regional endpoint resolver to regional or legacy `https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html`

**credentials**
Optional credentials shorthand for the config parameter

• `creds`:
  – `access_key_id`: AWS access key ID
  – `secret_access_key`: AWS secret access key
  – `session_token`: AWS temporary session token

• `profile`: The name of a profile to use. If not given, then the default profile is used.

• `anonymous`: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**
A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- ec2instanceconnect(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    )
  )
)
```
Operations

send_serial_console_ssh_public_key  
 Pushes an SSH public key to the specified EC2 instance

send_ssh_public_key  
 Pushes an SSH public key to the specified EC2 instance for use by the specified user

Examples

```r
## Not run:
svc <- ec2instanceconnect()
# The following example pushes a sample SSH public key to the EC2 instance
# i-abcd1234 in AZ us-west-2b for use by the instance OS user ec2-user.
svc$send_ssh_public_key(
  AvailabilityZone = "us-west-2a",
  InstanceId = "i-abcd1234",
  InstanceOSUser = "ec2-user",
  SSHPublicKey = "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAAABAQC3FlHqj2eqCdrGHuA6d..."
)
## End(Not run)
```

ecr  
Amazon EC2 Container Registry

Description

Amazon Elastic Container Registry

Amazon Elastic Container Registry (Amazon ECR) is a managed container image registry service. Customers can use the familiar Docker CLI, or their preferred client, to push, pull, and manage images. Amazon ECR provides a secure, scalable, and reliable registry for your Docker or Open Container Initiative (OCI) images. Amazon ECR supports private repositories with resource-based permissions using IAM so that specific users or Amazon EC2 instances can access repositories and images.

Amazon ECR has service endpoints in each supported Region. For more information, see Amazon ECR endpoints in the Amazon Web Services General Reference.
Usage

ecr(config = list(), credentials = list(), endpoint = NULL, region = NULL)

Arguments

cfg Optional configuration of credentials, endpoint, and/or region.

- credentials:
  - creds:
    - access_key_id: AWS access key ID
    - secret_access_key: AWS secret access key
    - session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.

- endpoint: The complete URL to use for the constructed client.

- region: The AWS Region used in instantiating the client.

- close_connection: Immediately close all HTTP connections.

- timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

- s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

- sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter

- creds:
  - access_key_id: AWS access key ID
  - secret_access_key: AWS secret access key
  - session_token: AWS temporary session token

- profile: The name of a profile to use. If not given, then the default profile is used.

- anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.
**Service syntax**

```r
svc <- ecr(
  config = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string",
  close_connection = "logical",
  timeout = "numeric",
  s3_force_path_style = "logical",
  sts_regional_endpoint = "string"
),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

**Operations**

- **batch_check_layer_availability**: Checks the availability of one or more image layers in a repository
- **batch_delete_image**: Deletes a list of specified images within a repository
- **batch_get_image**: Gets detailed information for an image
- **batch_get_repository_scanning_configuration**: Gets the scanning configuration for one or more repositories
- **complete_layer_upload**: Informs Amazon ECR that the image layer upload has completed for a specified registry, repository name, and upload ID
- **create_pull_through_cache_rule**: Creates a pull through cache rule
- **create_repository**: Creates a repository
- **delete_lifecycle_policy**: Deletes the lifecycle policy associated with the specified repository
- **delete_pull_through_cache_rule**: Deletes a pull through cache rule
- **delete_registry_policy**: Deletes the registry permissions policy
- **delete_repository**: Deletes a repository
- **delete_repository_policy**: Deletes the repository policy associated with the specified repository
- **describe_image_replication_status**: Returns the replication status for a specified image
- **describe_images**: Returns metadata about the images in a repository
describe_image_scan_findings
describe_pull_through_cache_rules
get_authorization_token
describe_registry
describe_repositories
get_download_url_for_layer
get_lifecycle_policy
get_lifecycle_policy_preview
get_registry_policy
get_registry_scanning_configuration
get_replication_configuration
get_registry_policy_preview
initiate_layer_upload
list_images
list_tags_for_resource
put_image
put_image_scanning_configuration
put_image_tag_mutability
put_lifecycle_policy
put_registry_policy
put_registry_scanning_configuration
put_registry_policy_preview
put_replication_configuration
set_repository_policy
start_image_scan
start_lifecycle_policy_preview
tag_resource
untag_resource
update_pull_through_cache_rule
upload_layer_part
validate_pull_through_cache_rule

Returns the scan findings for the specified image
Returns the pull through cache rules for a registry
Describes the settings for a registry
Describes image repositories in a registry
Retrieves an authorization token
Retrieves the pre-signed Amazon S3 download URL corresponding to an image
Retrieves the lifecycle policy for the specified repository
Retrieves the results of the lifecycle policy preview request for the specified repository
Retrieves the permissions policy for a registry
Retrieves the scanning configuration for a registry
Retrieves the repository policy for the specified repository
Notifies Amazon ECR that you intend to upload an image layer
Lists all the image IDs for the specified repository
List the tags for an Amazon ECR resource
Creates or updates the image manifest and tags associated with an image
The PutImageScanningConfiguration API is being deprecated, in favor of specifying the image scanning configuration at the registry level
Updates the image tag mutability settings for the specified repository
Creates or updates the lifecycle policy for the specified repository
Creates or updates the permissions policy for your registry
Creates or updates the scanning configuration for your private registry
Creates or updates the replication configuration for a registry
Applies a repository policy to the specified repository to control access permissions
Starts an image vulnerability scan
Starts a preview of a lifecycle policy for the specified repository
Adds specified tags to a resource with the specified ARN
Deletes specified tags from a resource
Updates an existing pull through cache rule
Uploads an image layer part to Amazon ECR
Validates an existing pull through cache rule for an upstream registry that requires authentication

Examples

```r
## Not run:
svc <- ecr()
# This example deletes images with the tags precise and trusty in a
# repository called ubuntu in the default registry for an account.
svc$batch_delete_image(
  imageIds = list(
    list(
      imageTag = "precise"
    ),
    repositoryName = "ubuntu"
  ),
  repositoryName = "ubuntu"
)
```

## End(Not run)
Amazon Elastic Container Registry Public (Amazon ECR Public) is a managed container image registry service. Amazon ECR provides both public and private registries to host your container images. You can use the Docker CLI or your preferred client to push, pull, and manage images. Amazon ECR provides a secure, scalable, and reliable registry for your Docker or Open Container Initiative (OCI) images. Amazon ECR supports public repositories with this API. For information about the Amazon ECR API for private repositories, see Amazon Elastic Container Registry API Reference.

Usage

```r
ecrpublic(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

- **credentials**: Optional credentials shorthand for the config parameter
• creds:
  – access_key_id: AWS access key ID
  – secret_access_key: AWS secret access key
  – session_token: AWS temporary session token
• profile: The name of a profile to use. If not given, then the default profile is used.
• anonymous: Set anonymous credentials.

endpoint  Optional shorthand for complete URL to use for the constructed client.
region    Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

svc <- ecrpublic(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
Operations

- **batch_check_layer_availability**: Checks the availability of one or more image layers that are within a repository in a public registry.
- **batch_delete_image**: Deletes a list of specified images that are within a repository in a public registry.
- **complete_layer_upload**: Informs Amazon ECR that the image layer upload is complete for a specified public registry, repository, and upload ID.
- **create_repository**: Creates a repository in a public registry.
- **delete_repository**: Deletes a repository in a public registry.
- **delete_repository_policy**: Deletes the repository policy that’s associated with the specified repository.
- **describe_images**: Returns metadata that’s related to the images in a repository in a public registry.
- **describe_image_tags**: Returns the image tag details for a repository in a public registry.
- **describe_registries**: Returns details for a public registry.
- **describe_repositories**: Describes repositories that are in a public registry.
- **get_authorization_token**: Retrieves an authorization token.
- **get_registry_catalog_data**: Retrieves catalog metadata for a public registry.
- **get_repository_catalog_data**: Retrieve catalog metadata for a repository in a public registry.
- **get_repository_policy**: Retrieves the repository policy for the specified repository.
- **initiate_layer_upload**: Notifies Amazon ECR that you intend to upload an image layer.
- **list_tags_for_resource**: List the tags for an Amazon ECR Public resource.
- **put_image**: Creates or updates the image manifest and tags that are associated with an image.
- **put_registry_catalog_data**: Create or update the catalog data for a public registry.
- **put_repository_catalog_data**: Creates or updates the catalog data for a repository in a public registry.
- **set_repository_policy**: Applies a repository policy to the specified public repository to control access permissions.
- **tag_resource**: Associates the specified tags to a resource with the specified resourceArn.
- **untag_resource**: Deletes specified tags from a resource.
- **upload_layer_part**: Uploads an image layer part to Amazon ECR.

Examples

```r
## Not run:
svc <- ecrpublic()
svc$batch_check_layer_availability(
  Foo = 123
)

## End(Not run)
```

---

**ecs**

*Amazon EC2 Container Service*

**Description**

Amazon Elastic Container Service

Amazon Elastic Container Service (Amazon ECS) is a highly scalable, fast, container management service. It makes it easy to run, stop, and manage Docker containers. You can host your cluster on
a serverless infrastructure that’s managed by Amazon ECS by launching your services or tasks on Fargate. For more control, you can host your tasks on a cluster of Amazon Elastic Compute Cloud (Amazon EC2) or External (on-premises) instances that you manage.

Amazon ECS makes it easy to launch and stop container-based applications with simple API calls. This makes it easy to get the state of your cluster from a centralized service, and gives you access to many familiar Amazon EC2 features.

You can use Amazon ECS to schedule the placement of containers across your cluster based on your resource needs, isolation policies, and availability requirements. With Amazon ECS, you don’t need to operate your own cluster management and configuration management systems. You also don’t need to worry about scaling your management infrastructure.

Usage

```r
ecs(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

- **endpoint**: Optional shorthand for complete URL to use for the constructed client.

- **region**: Optional shorthand for AWS Region used in instantiating the client.
Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

svc <- ecs(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)

Operations

- create_capacity_provider: Creates a new capacity provider
- create_cluster: Creates a new Amazon ECS cluster
- create_service: Runs and maintains your desired number of tasks from a specified task definition
- create_task_set: Create a task set in the specified cluster and service
- delete_account_setting: Disables an account setting for a specified user, role, or the root user for an account
- delete_attributes: Deletes one or more custom attributes from an Amazon ECS resource
- delete_capacity_provider: Deletes the specified capacity provider
- delete_cluster: Deletes the specified cluster
delete_service 
delete_task_definitions 
delete_task_set 
deregister_container_instance 
deregister_task_definition 
describe_capacity_providers 
describe_clusters 
describe_container_instances 
describe_services 
describe_task_definition 
describe_tasks 
describe_task_sets 
discover_poll_endpoint 
execute_command 
get_task_protection 
lst_account_settings 
lst_attributes 
lst_clusters 
lst_container_instances 
lst_services 
lst_services_by_namespace 
lst_tags_for_resource 
lst_task_definition_families 
lst_task_definitions 
lst_tasks 
put_account_setting 
put_account_setting_default 
put_attributes 
put_cluster_capacity_providers 
register_container_instance 
register_task_definition 
run_task 
start_task 
stop_task 
submit_attachment_state_changes 
submit_container_state_change 
submit_task_state_change 
tag_resource 
untag_resource 
update_capacity_provider 
update_cluster 
update_cluster_settings 
update_container_agent 
update_container_instances_state 
update_service 
update_service_primary_task_set 
update_task_protection 
update_task_set 

Deletes a specified service within a cluster
Deletes one or more task definitions
Deletes a specified task set within a service
Deregisters an Amazon ECS container instance from the specified cluster
Deregisters the specified task definition by family and revision
Describes one or more of your capacity providers
Describes one or more of your clusters
Describes one or more container instances
Describes the specified services running in your cluster
Describes a task definition
Describes a specified task or tasks
Describes the task sets in the specified cluster and service
This action is only used by the Amazon ECS agent, and it is not intended for use outside of the agent
Runs a command remotely on a container within a task
Retrieves the protection status of tasks in an Amazon ECS service
Lists the account settings for a specified principal
Lists the attributes for Amazon ECS resources within a specified target type and cluster
Returns a list of existing clusters
Returns a list of container instances in a specified cluster
Returns a list of services
This operation lists all of the services that are associated with a Cloud Map namespace
List the tags for an Amazon ECS resource
Returns a list of task definition families that are registered to your account
Returns a list of task definitions that are registered to your account
Returns a list of tasks
Modifies an account setting
Modifies an account setting for all users on an account for whom no individual account setting is defined
Create or update an attribute on an Amazon ECS resource
Modifies the available capacity providers and the default capacity provider strategy for a cluster
This action is only used by the Amazon ECS agent, and it is not intended for use outside of the agent
Registers a new task definition from the supplied family and containerDefinitions
Starts a new task using the specified task definition
Starts a new task from the specified task definition on the specified container instance or instances
Stops a running task
This action is only used by the Amazon ECS agent, and it is not intended for use outside of the agent
This action is only used by the Amazon ECS agent, and it is not intended for use outside of the agent
This action is only used by the Amazon ECS agent, and it is not intended for use outside of the agent
 Associates the specified tags to a resource with the specified resourceArn
Deletes specified tags from a resource
Modifies the parameters for a capacity provider
Modifies the settings to use for a cluster
Updates the cluster
Updates the Amazon ECS container agent on a specified container instance
Modifies the status of an Amazon ECS container instance
Modifies the parameters of a service
Modifies which task set in a service is the primary task set
Updates the protection status of a task
Modifies a task set
Examples

```r
## Not run:
svc <- ecs()
# This example creates a cluster in your default region.
svc$create_cluster(
  clusterName = "my_cluster"
)
## End(Not run)
```

Description

Amazon Elastic File System (Amazon EFS) provides simple, scalable file storage for use with Amazon EC2 Linux and Mac instances in the Amazon Web Services Cloud. With Amazon EFS, storage capacity is elastic, growing and shrinking automatically as you add and remove files, so that your applications have the storage they need, when they need it. For more information, see the Amazon Elastic File System API Reference and the Amazon Elastic File System User Guide.

Usage

```r
esfs(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.

• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

**credentials**

Optional credentials shorthand for the `config` parameter

• **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

**endpoint**

Optional shorthand for complete URL to use for the constructed client.

**region**

Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- efs(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    )
  )
)
```
Operations

create_access_point
create_file_system
create_mount_target
create_replication_configuration
create_tags
delete_access_point
delete_file_system
delete_file_system_policy
delete_mount_target
delete_replication_configuration
delete_tags
describe_access_points
describe_account_preferences
describe_backup_policy
describe_file_system_policy
describe_file_systems
describe_lifecycle_configuration
describe_mount_targets
describe_mount_target_security_groups
describe_replication_configurations
describe_tags
list_tags_for_resource
modify_mount_target_security_groups
put_account_preferences
put_backup_policy
put_file_system_policy
put_lifecycle_configuration
tag_resource
untag_resource
update_file_system
update_file_system_protection

create_access_point
create_file_system
create_mount_target
create_replication_configuration
create_tags
delete_access_point
delete_file_system
delete_file_system_policy
delete_mount_target
delete_replication_configuration
delete_tags
describe_access_points
describe_account_preferences
describe_backup_policy
describe_file_system_policy
describe_file_systems
describe_lifecycle_configuration
describe_mount_targets
describe_mount_target_security_groups
describe_replication_configurations
describe_tags
list_tags_for_resource
modify_mount_target_security_groups
put_account_preferences
put_backup_policy
put_file_system_policy
put_lifecycle_configuration
tag_resource
untag_resource
update_file_system
update_file_system_protection

Examples

## Not run:
svc <- efs()
This operation creates a new, encrypted file system with automatic backups enabled, and the default general-purpose performance mode.

```r
svc$create_file_system(
  Backup = TRUE,
  CreationToken = "tokenstring",
  Encrypted = TRUE,
  PerformanceMode = "generalPurpose",
  Tags = list(
    list(
      Key = "Name",
      Value = "MyFileSystem"
    )
  )
)
```

## End(Not run)

### eks

**Amazon Elastic Kubernetes Service**

**Description**

Amazon Elastic Kubernetes Service (Amazon EKS) is a managed service that makes it easy for you to run Kubernetes on Amazon Web Services without needing to setup or maintain your own Kubernetes control plane. Kubernetes is an open-source system for automating the deployment, scaling, and management of containerized applications.

Amazon EKS runs up-to-date versions of the open-source Kubernetes software, so you can use all the existing plugins and tooling from the Kubernetes community. Applications running on Amazon EKS are fully compatible with applications running on any standard Kubernetes environment, whether running in on-premises data centers or public clouds. This means that you can easily migrate any standard Kubernetes application to Amazon EKS without any code modification required.

**Usage**

```r
eks(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

**Arguments**

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
– **anonymous**: Set anonymous credentials.

- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.

**credentials**

Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint**

Optional shorthand for complete URL to use for the constructed client.

**region**

Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- eks(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
  ),
)```
sts_regional_endpoint = "string"
),
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

associate_access_policy Associates an access policy and its scope to an access entry
associate_encryption_config Associates an encryption configuration to an existing cluster
associate_identity_provider_config Associates an identity provider configuration to a cluster
create_access_entry Creates an access entry
create_addon Creates an Amazon EKS add-on
create_cluster Creates an Amazon EKS control plane
create_eks_anywhere_subscription Creates an EKS Anywhere subscription
create_fargate_profile Creates a managed node group for an Amazon EKS cluster
create_pod_identity_association Creates an EKS Pod Identity association between a service account in an Amazon EKS cluster and an IAM role with EKS Pod Identity
delete_access_entry Deletes an access entry
delete_addon Deletes an Amazon EKS add-on
delete_cluster Deletes an Amazon EKS cluster control plane
delete_nodegroup Deletes an expired or inactive subscription
delete_fargate_profile Deletes an Fargate profile
delete_pod_identity_association Deletes a managed node group
deregister_cluster Deregisters a connected cluster to remove it from the Amazon EKS control plane
describe_access_entry Describes an access entry
describe_addon Describes an Amazon EKS add-on
describe_addon_configuration Returns configuration options
describe_addon_versions Describes the versions for an add-on
describe_cluster Describes an Amazon EKS cluster
describe_eks_anywhere_subscription Returns descriptive information about a subscription
describe_fargate_profile Describes a Fargate profile
describe_identity_provider_config Describes an identity provider configuration
describe_insight Returns details about an insight that you specify using its ID
describe_nodegroup Describes a managed node group
describe_pod_identity_association Returns descriptive information about an EKS Pod Identity association
describe_update Describes an update to an Amazon EKS resource
disassociate_access_policy Disassociates an access policy from an access entry
### Examples

```r
## Not run:
svc <- eks()

# The following example creates an Amazon EKS cluster called prod.
svc$create_cluster(
  version = "1.10",
  name = "prod",
  clientRequestToken = "1d2129a1-3d38-460a-9756-e5b91fdd951",
  resourcesVpcConfig = list(
    securityGroupIds = list("sg-6979fe18"),
    subnetIds = list("subnet-6782e71e", "subnet-e7e761ac")
  ),
  roleArn = "arn:aws:iam::012345678910:role/eks-service-role-AWSServiceRole..."
)
```

## End(Not run)
**Description**

Amazon ElastiCache is a web service that makes it easier to set up, operate, and scale a distributed cache in the cloud.

With ElastiCache, customers get all of the benefits of a high-performance, in-memory cache with less of the administrative burden involved in launching and managing a distributed cache. The service makes setup, scaling, and cluster failure handling much simpler than in a self-managed cache deployment.

In addition, through integration with Amazon CloudWatch, customers get enhanced visibility into the key performance statistics associated with their cache and can receive alarms if a part of their cache runs hot.

**Usage**

```r
elasticache(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

**Arguments**

- **config**: Optional configuration of credentials, endpoint, and/or region.

  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  
  - **endpoint**: The complete URL to use for the constructed client.
  
  - **region**: The AWS Region used in instantiating the client.
  
  - **close_connection**: Immediately close all HTTP connections.
  
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  
  - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.  

• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy. [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials** Optional credentials shorthand for the config parameter

• **creds**:
  – **access_key_id**: AWS access key ID
  – **secret_access_key**: AWS secret access key
  – **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- elasticache(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string",
  close_connection = "logical",
  timeout = "numeric",
  s3_force_path_style = "logical",
  sts_regional_endpoint = "string"
)
```
Anonymous = "logical"
",
endpoint = "string",
region = "string"
)

Operations

add_tags_to_resource
authorize_cache_security_group_ingress
batch_apply_update_action
batch_stop_update_action
complete_migration
copy_serverless_cache_snapshot
copy_snapshot
create_cache_cluster
create_cache_parameter_group
create_cache_security_group
create_cache_subnet_group
create_global_replication_group
create_replication_group
create_serverless_cache
create_serverless_cache_snapshot
create_snapshot
create_user
create_user_group
decrease_node_groups_in_global_replication_group
decrease_replica_count
delete_cache_cluster
delete_cache_parameter_group
delete_cache_security_group
delete_cache_subnet_group
delete_global_replication_group
delete_replication_group
delete_serverless_cache
delete_serverless_cache_snapshot
delete_snapshot
delete_user
delete_user_group
describe_cache_clusters
describe_cache_engine_versions
describe_cache_parameter_groups
describe_cache_parameters
describe_cache_security_groups
describe_cache_subnet_groups
describe_engine_default_parameters
describe_events
describe_global_replication_groups

A tag is a key-value pair where the key and value are case-sensitive
Allows network ingress to a cache security group
Apply the service update
Stop the service update
Complete the migration of data
Creates a copy of an existing serverless cache's snapshot
Makes a copy of an existing snapshot
Creates a cluster
Creates a new Amazon ElastiCache cache parameter group
Creates a new cache security group
Creates a new cache subnet group
Global Datastore for Redis offers fully managed, fast, reliable and secure
Creates a Redis (cluster mode disabled) or a Redis (cluster mode enabled)
Creates a serverless cache
This API creates a copy of an entire ServerlessCache at a specific moment
Creates a copy of an entire cluster or replication group at a specific moment
For Redis engine version 6
For Redis engine version 6
Decreases the number of node groups in a Global datastore
Dynamically decreases the number of replicas in a Redis (cluster mode disabled)
Deletes a previously provisioned cluster
Deletes the specified cache parameter group
Deletes a cache security group
Deletes a cache subnet group
Deleting a Global datastore is a two-step process:
Deletes an existing replication group
Deletes a specified existing serverless cache
Deletes an existing serverless cache snapshot
Deletes an existing snapshot
For Redis engine version 6
For Redis engine version 6
Returns information about all provisioned clusters if no cluster identifier is specified
Returns a list of the available cache engines and their versions
Returns a list of cache parameter group descriptions
Returns the detailed parameter list for a particular cache parameter group
Returns a list of cache security group descriptions
Returns a list of cache subnet group descriptions
Returns the default engine and system parameter information for the specified
Returns events related to clusters, cache security groups, and cache parameter
describe_replication_groups
describe_reserved_cache_nodes
describe_reserved_cache_nodes_offerings
describe_serverless_caches
describe_serverless_cache_snapshots
describe_service_updates
describe_snapshots
describe_update_actions
describe_user_groups
describe_users
disassociate_global_replication_group
export_serverless_cache_snapshot
failover_global_replication_group
increase_node_groups_in_global_replication_group
increase_replica_count
list_allowed_node_type_modifications
list_tags_for_resource
modify_cache_cluster
modify_cache_parameter_group
modify_cache_subnet_group
modify_global_replication_group
modify_replication_group
modify_replication_group_shard_configuration
modify_serverless_cache
modify_user
modify_user_group
purchase_reserved_cache_nodes_offering
rebalance_slots_in_global_replication_group
reboot_cache_cluster
remove_tags_from_resource
reset_cache_parameter_group
revoke_cache_security_group_ingress
start_migration
test_failover
test_migrate

test_migration

Examples

## Not run:
svc <- elasticache()
svc$add_tags_to_resource(
   Foo = 123
)

## End(Not run)
Description

AWS Elastic Beanstalk makes it easy for you to create, deploy, and manage scalable, fault-tolerant applications running on the Amazon Web Services cloud.

For more information about this product, go to the AWS Elastic Beanstalk details page. The location of the latest AWS Elastic Beanstalk WSDL is https://elasticbeanstalk.s3.amazonaws.com/doc/2010-12-01/AWSElasticBeanstalk.wsdl. To install the Software Development Kits (SDKs), Integrated Development Environment (IDE) Toolkits, and command line tools that enable you to access the API, go to Tools for Amazon Web Services.

Endpoints

For a list of region-specific endpoints that AWS Elastic Beanstalk supports, go to Regions and Endpoints in the Amazon Web Services Glossary.

Usage

elasticbeanstalk(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

config Optional configuration of credentials, endpoint, and/or region.

• credentials:
  • creds:
    • access_key_id: AWS access key ID
    • secret_access_key: AWS secret access key
    • session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

• endpoint: The complete URL to use for the constructed client.
• region: The AWS Region used in instantiating the client.
• close_connection: Immediately close all HTTP connections.
• timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials** Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.

- **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- elasticbeanstalk(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  )
)
```
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

abort_environment_update
apply_environment_managed_action
associate_environment_operations_role
check_dns_availability
compose_environments
create_application
create_application_version
create_configuration_template
create_environment
create_platform_version
create_storage_location
delete_application
delete_application_version
delete_configuration_template
delete_environment_configuration
delete_platform_version
describe_account_attributes
describe_applications
describe_application_versions
describe_configuration_options
describe_configuration_settings
describe_environment_health
describe_environment_managed_action_history
describe_environment_managed_actions
describe_environment_resources
describe_environments
describe_events
describe_instances_health
describe_platform_version
disassociate_environment_operations_role
list_available_solution_stacks
list_platform_branches
list_platform_versions
list_tags_for_resource
rebuild_environment
request_environment_info
restart_app_server
retrieve_environment_info
swap_environment_cnam_es
terminate_environment

Cancels in-progress environment configuration update or application version deployment
Applies a scheduled managed action immediately
Add or change the operations role used by an environment
Checks if the specified CNAME is available
Create or update a group of environments that each run a separate component of an application
Creates an application that has one configuration template named default and no application versions
Creates an application version for the specified application
Create an AWS Elastic Beanstalk configuration template, associated with a specified solution stack
Launches an AWS Elastic Beanstalk environment for the specified application
Create a new version of your custom platform
Creates a bucket in Amazon S3 to store application versions, logs, and other files used by Elastic Beanstalk environments
Deletes the specified application along with all associated versions and configuration files
Deletes the specified version from the specified application
Deletes the specified configuration template
Deletes the draft configuration associated with the running environment
Deletes the specified version of a custom platform
Returns attributes related to AWS Elastic Beanstalk that are associated with the calling AWS account
Returns the descriptions of existing applications
Retrieve a list of application versions
Describes the configuration options that are used in a particular configuration template or environment
Returns a description of the settings for the specified configuration set, that is, the configuration template or the configuration set associated with the running environment
Returns information about the overall health of the specified environment
Lists an environment’s completed and failed managed actions
Lists an environment’s upcoming and in-progress managed actions
Returns AWS resources for this environment
Returns descriptions for existing environments
Returns list of event descriptions matching criteria up to the last 6 weeks
Retrieves detailed information about the health of instances in your AWS Elastic Beanstalk environment
Describes a platform version
Disassociate the operations role from an environment
Returns a list of the available solution stack names, with the public version first and then in reverse chronological order
Lists the platform branches available for your account in an AWS Region
Lists the platform versions available for your account in an AWS Region
Return the tags applied to an AWS Elastic Beanstalk resource
Deletes and recreates all of the AWS resources (for example: the Auto Scaling group, load balancer, etc)
Initiates a request to compile the specified type of information of the deployment
Causes the environment to restart the application container server running on each Amazon EC2 instance
Retrieves the compiled information from a RequestEnvironmentInfo request
Swaps the CNAMEs of two environments
Terminates the specified environment
Examples

```r
## Not run:
svc <- elasticbeanstalk()
# The following code aborts a running application version deployment for
# an environment named my-env:
svc$abort_environment_update(
  EnvironmentName = "my-env"
)

## End(Not run)
```

elasticinference  Amazon Elastic Inference

Description

Elastic Inference public APIs.

February 15, 2023: Starting April 15, 2023, AWS will not onboard new customers to Amazon Elastic Inference (EI), and will help current customers migrate their workloads to options that offer better price and performance. After April 15, 2023, new customers will not be able to launch instances with Amazon EI accelerators in Amazon SageMaker, Amazon ECS, or Amazon EC2. However, customers who have used Amazon EI at least once during the past 30-day period are considered current customers and will be able to continue using the service.

Usage

```r
elasticinference(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```
Arguments

config Optional configuration of credentials, endpoint, and/or region.

- credentials:
  - creds:
    * access_key_id: AWS access key ID
    * secret_access_key: AWS secret access key
    * session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.

- endpoint: The complete URL to use for the constructed client.

- region: The AWS Region used in instantiating the client.

- close_connection: Immediately close all HTTP connections.

- timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

- s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

- sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html

credentials Optional credentials shorthand for the config parameter

- creds:
  - access_key_id: AWS access key ID
  - secret_access_key: AWS secret access key
  - session_token: AWS temporary session token

- profile: The name of a profile to use. If not given, then the default profile is used.

- anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```
svc <- elasticinference(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
```
elasticinference

secret_access_key = "string",
session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string",
close_connection = "logical",
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
creds = list(
access_key_id = "string",
secret_access_key = "string",
session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

describe_accelerator_offerings Describes the locations in which a given accelerator type or set of types is present in a given region

describe_accelerators Describes information over a provided set of accelerators belonging to an account

describe_accelerator_types Describes the accelerator types available in a given region, as well as their characteristics, such as memory and throughput

describe_accelerator_types
list_tags_for_resource Returns all tags of an Elastic Inference Accelerator

tag_resource Adds the specified tags to an Elastic Inference Accelerator

Examples

## Not run:
svc <- elasticinference()
svc$describe_accelerator_offerings(
   Foo = 123
)

## End(Not run)
Description

Amazon Elasticsearch Configuration Service

Use the Amazon Elasticsearch Configuration API to create, configure, and manage Elasticsearch domains.

For sample code that uses the Configuration API, see the Amazon Elasticsearch Service Developer Guide. The guide also contains sample code for sending signed HTTP requests to the Elasticsearch APIs.

The endpoint for configuration service requests is region-specific: es.<region>.amazonaws.com. For example, es.us-east-1.amazonaws.com. For a current list of supported regions and endpoints, see Regions and Endpoints.

Usage

```
elasticsearchservice(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  
  - **credentials**:
    
    - **creds**:
      
      * **access_key_id**: AWS access key ID
      * **secret_access_key**: AWS secret access key
      * **session_token**: AWS temporary session token
    
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    
    - **anonymous**: Set anonymous credentials.
  
  - **endpoint**: The complete URL to use for the constructed client.
  
  - **region**: The AWS Region used in instantiating the client.
  
  - **close_connection**: Immediately close all HTTP connections.
  
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  
  - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  
credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value
A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax
```
svc <- elasticsearchservice(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
```
region = "string"
)

## Operations

- `accept_inbound_cross_cluster_search_connection` allows the destination domain owner to accept an inbound cross-cluster search connection request.
- `add_tags` attaches tags to an existing Elasticsearch domain.
- `associate_package` associates a package with an Amazon ES domain.
- `authorize_vpc_endpoint_access` provides access to an Amazon OpenSearch Service domain through a VPC endpoint.
- `cancel_domain_config_change` cancels a pending configuration change on an Amazon OpenSearch Service domain.
- `cancel_elasticsearch_service_software_update` cancels a scheduled service software update for an Amazon ES domain.
- `create_elasticsearch_domain` creates a new Elasticsearch domain.
- `create_cross_cluster_search_connection` creates a new cross-cluster search connection from a source domain to a destination domain.
- `create_package` creates a package for use with Amazon ES domains.
- `create_vpc_endpoint` creates an Amazon OpenSearch Service-managed VPC endpoint.
- `delete_elasticsearch_domain` permanently deletes the specified Elasticsearch domain and all of its data.
- `delete_elasticsearch_service_role` deletes the service-linked role that Elasticsearch Service uses to manage and maintain VPC domains.
- `delete_inbound_cross_cluster_search_connection` allows the destination domain owner to delete an existing inbound cross-cluster search connection.
- `delete_outbound_cross_cluster_search_connection` allows the source domain owner to delete an existing outbound cross-cluster search connection.
- `delete_package` deletes the package.
- `delete_vpc_endpoint` deletes an Amazon OpenSearch Service-managed interface VPC endpoint.
- `describe_domain_auto_tunes` returns information about the scheduled Auto-Tune action for the Elasticsearch domain, such as Auto-Tune action type, description, severity, and scheduled date.
- `describe_domain_change_progress` returns information about the current blue/green deployment happening on a domain, including change ID, status, and progress stages.
- `describe_elasticsearch_domain` returns domain configuration information about the specified Elasticsearch domain, including the domain ID, domain endpoint, and domain ARN.
- `describe_elasticsearch_domain_config` provides cluster configuration information about the specified Elasticsearch domain, such as the state, creation date, update version, and update date for cluster options.
- `describe_elasticsearch_domains` returns domain configuration information about the specified Elasticsearch domains, including the domain ID, domain endpoint, and domain ARN.
- `describe_elasticsearch_instance_type_limits` describes Elasticsearch Limits for a given InstanceType and ElasticsearchVersion.
- `describe_inbound_cross_cluster_search_connections` lists all the inbound cross-cluster search connections for a destination domain.
- `describe_outbound_cross_cluster_search_connections` lists all the outbound cross-cluster search connections for a source domain.
- `describe_packages` describes all packages available to Amazon ES.
- `describe_reserved_elasticsearch_instance_offering` lists available reserved Elasticsearch instance offerings.
- `describe_reserved_elasticsearch_instances` returns information about reserved Elasticsearch instances for this account.
- `describe_vpc_endpoints` describes one or more Amazon OpenSearch Service-managed VPC endpoints.
- `describe_vpc_endpoint_access` describes the principal that is allowed to access an Amazon OpenSearch Service-managed VPC endpoint.
- `describe_vpc_endpoints` lists all Amazon OpenSearch Service-managed VPC endpoints in the current account and Region.
- `describe_vpc_endpoints_for_domain` lists all Amazon OpenSearch Service-managed VPC endpoints associated with a particular domain.
- `get_compatible_elasticsearch_versions` returns a list of upgrade compatible Elasticsearch versions.
- `get_package_version_history` retrieves the complete history of the last 10 upgrades that were performed on the domain.
- `get_upgrade_history` retrieves the latest status of the last upgrade or upgrade eligibility check that was performed on the domain.
- `get_upgrade_status` retrieves the latest status of the last upgrade or upgrade eligibility check that was performed on the domain.
- `get_vpc_endpoint_access` retrieves all Amazon OpenSearch Service-managed VPC endpoints associated with the principal.
- `get_vpc_endpoints` retrieves all Amazon OpenSearch Service-managed VPC endpoints associated with the domain.
- `get_vpc_endpoints_for_domain` retrieves all Amazon OpenSearch Service-managed VPC endpoints associated with a particular domain.
- `list_domain_names` returns the name of all Elasticsearch domains owned by the current user.
- `list_domains_for_package` lists all Elasticsearch domains associated with the package.
- `list_elasticsearch_instance_types` lists all Elasticsearch instance types that are supported for the specified Elasticsearch version.
- `list_elasticsearch_versions` lists all supported Elasticsearch versions.
- `list_packages_for_domain` lists all packages associated with the domain.
- `list_vpc_endpoint_access` retrieves all Amazon OpenSearch Service-managed VPC endpoints associated with the principal.
- `list_vpc_endpoints` retrieves all Amazon OpenSearch Service-managed VPC endpoints associated with the domain.
- `list_vpc_endpoints_for_domain` retrieves all Amazon OpenSearch Service-managed VPC endpoints associated with a particular domain.
- `list_tags` retrieves all tags for the given Elasticsearch domain.
- `purchase_reserved_elasticsearch_instance_offering` allows you to purchase reserved Elasticsearch instances.
Elastic Load Balancing

Description

A load balancer can distribute incoming traffic across your EC2 instances. This enables you to increase the availability of your application. The load balancer also monitors the health of its registered instances and ensures that it routes traffic only to healthy instances. You configure your load balancer to accept incoming traffic by specifying one or more listeners, which are configured with a protocol and port number for connections from clients to the load balancer and a protocol and port number for connections from the load balancer to the instances.

Elastic Load Balancing supports three types of load balancers: Application Load Balancers, Network Load Balancers, and Classic Load Balancers. You can select a load balancer based on your application needs. For more information, see the Elastic Load Balancing User Guide.

This reference covers the 2012-06-01 API, which supports Classic Load Balancers. The 2015-12-01 API supports Application Load Balancers and Network Load Balancers.

To get started, create a load balancer with one or more listeners using create_load_balancer. Register your instances with the load balancer using register_instances_with_load_balancer. All Elastic Load Balancing operations are idempotent, which means that they complete at most one time. If you repeat an operation, it succeeds with a 200 OK response code.

Usage

```r
elb(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```
Arguments

Optional configuration of credentials, endpoint, and/or region.

- config

  • credentials:
    - creds:
      - access_key_id: AWS access key ID
      - secret_access_key: AWS secret access key
      - session_token: AWS temporary session token
    - profile: The name of a profile to use. If not given, then the default profile is used.
    - anonymous: Set anonymous credentials.

  • endpoint: The complete URL to use for the constructed client.

  • region: The AWS Region used in instantiating the client.

  • close_connection: Immediately close all HTTP connections.

  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoint.html

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```
svc <- elb(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
      ),
    ),
  ),
).
```
secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
),
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

add_tags
apply_security_groups_to_load_balancer
attach_load_balancer_to_subnets
configure_health_check
create_app_cookie_stickiness_policy
create_lb_cookie_stickiness_policy
create_load_balancer
create_load_balancer_listeners
create_load_balancer_policy
delete_load_balancer
delete_load_balancer_listeners
delete_load_balancer_policy
deregister_instances_from_load_balancer
describe_account_limits
describe_instance_health
describe_load_balancer_attributes
describe_load_balancer_policies
describe_load_balancer_policy_types
describe_load_balancers
describe_tags

Adds the specified tags to the specified load balancer
Associates one or more security groups with your load balancer in a virtual private cloud (VPC)
Adds one or more subnets to the set of configured subnets for the specified load balancer
Specifies the health check settings to use when evaluating the health state of your Amazon EC2 instances
Generates a stickiness policy with sticky session lifetimes that follow that of an application-generated cookie
Generates a stickiness policy with sticky session lifetimes controlled by the browser (user-agent) expiration period
Creates a Classic Load Balancer
Creates one or more listeners for the specified load balancer
Creates a policy with the specified attributes for the specified load balancer
Deletes the specified load balancer
Deletes the specified listeners from the specified load balancer
Deletes the specified policy from the specified load balancer
Deregisters the specified instances from the specified load balancer
Describes the current Elastic Load Balancing resource limits for your AWS account
Describes the state of the specified instances with respect to the specified load balancer
Describes the attributes for the specified load balancer
Describes the specified policies
Describes the specified load balancer policy types or all load balancer policy types
Describes the specified load balancers
Describes the tags associated with the specified load balancers
detach_load_balancer_from_subnets
disable_availability_zones_for_load_balancer
enable_availability_zones_for_load_balancer
modify_load_balancer_attributes
register_instances_with_load_balancer
remove_tags
set_load_balancer_listener_ssl_certificate
set_load_balancer_policies_for_backend_server
set_load_balancer_policies_of_listener

Examples

```r
## Not run:
svc <- elb()
# This example adds two tags to the specified load balancer.
svc$add_tags(
  LoadBalancerNames = list(
    "my-load-balancer"
  ),
  Tags = list(
    list(
      Key = "project",
      Value = "lima"
    ),
    list(
      Key = "department",
      Value = "digital-media"
    )
  )
)
## End(Not run)
```

elbv2

Elastic Load Balancing

Description

A load balancer distributes incoming traffic across targets, such as your EC2 instances. This enables you to increase the availability of your application. The load balancer also monitors the health of its registered targets and ensures that it routes traffic only to healthy targets. You configure your load balancer to accept incoming traffic by specifying one or more listeners, which are configured with a protocol and port number for connections from clients to the load balancer. You configure a target group with a protocol and port number for connections from the load balancer to the targets, and with health check settings to be used when checking the health status of the targets.
Elastic Load Balancing supports the following types of load balancers: Application Load Balancers, Network Load Balancers, Gateway Load Balancers, and Classic Load Balancers. This reference covers the following load balancer types:

- **Application Load Balancer** - Operates at the application layer (layer 7) and supports HTTP and HTTPS.
- **Network Load Balancer** - Operates at the transport layer (layer 4) and supports TCP, TLS, and UDP.
- **Gateway Load Balancer** - Operates at the network layer (layer 3).

For more information, see the Elastic Load Balancing User Guide.

All Elastic Load Balancing operations are idempotent, which means that they complete at most one time. If you repeat an operation, it succeeds.

### Usage

```python
elbv2(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

### Arguments

- **config**
  - Optional configuration of credentials, endpoint, and/or region.
  - **credentials**: 
    - **creds**: 
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-_endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-_endpoints.html)

- **credentials**
  - Optional credentials shorthand for the config parameter
  - **creds**: 
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
• **anonymous**: Set anonymous credentials.

  **endpoint**: Optional shorthand for complete URL to use for the constructed client.
  
  **region**: Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...),` where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- elbv2(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

**Operations**

- `add_listener_certificates`: Adds the specified SSL server certificate to the certificate list for the specified HTTPS or TLS listener.
- `add_tags`: Adds the specified tags to the specified Elastic Load Balancing resource.
- `add_trust_store_revocations`: Adds the specified revocation file to the specified trust store.
create_listener
create_load_balancer
create_rule
create_target_group
create_trust_store
delete_listener
delete_load_balancer
delete_rule
delete_target_group
delete_trust_store
deregister_targets
describe_account_limits
describe_listener_certificates
describe_listeners
describe_load_balancer_attributes
describe_load_balancers
describe_rules
describe_ssl_policies
describe_tags
describe_target_group_attributes
describe_target_groups
describe_target_health
describe_trust_store_associations
describe_trust_store_revolutions
describe_trust_stores
get_trust_store_ca_certificates_bundle
get_trust_store_revocation_content
modify_listener
modify_load_balancer_attributes
modify_rule
modify_target_group
modify_target_group_attributes
modify_trust_store
register_targets
remove_listener_certificates
remove_tags
remove_trust_store_revolutions
set_ip_address_type
set_rule_priorities
set_security_groups
set_subnets

create_listener
Create a listener for the specified Application Load Balancer, Network Load Balancer, or Gateway Load Balancer.
create_load_balancer
Create an Application Load Balancer, Network Load Balancer, or Gateway Load Balancer.
create_rule
Create a rule for the specified listener.
create_target_group
Create a target group.
create_trust_store
Create a trust store.
delete_listener
Delete the specified listener.
delete_load_balancer
Delete the specified Application Load Balancer, Network Load Balancer, or Gateway Load Balancer.
delete_rule
Delete the specified rule.
delete_target_group
Delete the specified target group.
delete_trust_store
Delete a trust store.
deregister_targets
Deregisters the specified targets from the specified target group.
describe_account_limits
Describe the current Elastic Load Balancing resource limits for your Amazon Web Services account.
describe_listener_certificates
Describe the default certificate and the certificate list for the specified HTTPS or TLS listener.
describe_listeners
Describe the specified listeners or the listeners for the specified Application Load Balancer, Network Load Balancer, or Gateway Load Balancer.
describe_load_balancer_attributes
Describe the attributes for the specified Application Load Balancer, Network Load Balancer, or Gateway Load Balancer.
describe_load_balancers
Describe the specified load balancers or all of your load balancers.
describe_rules
Describe the specified rules or the rules for the specified listener.
describe_ssl_policies
Describe the specified policies or all policies used for SSL negotiation.
describe_tags
Describe the tags for the specified Elastic Load Balancing resources.
describe_target_group_attributes
Describe the attributes for the specified target group.
describe_target_groups
Describe the specified target groups or all of your target groups.
describe_target_health
Describe the health of the specified targets or all of your targets.
describe_trust_store_associations
Describe all resources associated with the specified trust store.
describe_trust_store_revolutions
Describe the revocation files in use by the specified trust store arn, or revocation ID.
describe_trust_stores
Describe all trust stores for a given account by trust store arn’s or name.
get_trust_store_ca_certificates_bundle
Retrieve the ca certificate bundle.
get_trust_store_revocation_content
Retrieve the specified revocation file.
modify_listener
Replace the specified properties of the specified listener.
modify_load_balancer_attributes
Modify the specified attributes of the specified Application Load Balancer, Network Load Balancer, or Gateway Load Balancer.
modify_rule
Replace the specified properties of the specified rule.
modify_target_group
Modify the health checks used when evaluating the health state of the targets in the target group.
modify_target_group_attributes
Modify the specified attributes of the specified target group.
modify_trust_store
Update the ca certificate bundle for a given trust store.
register_targets
Register the specified targets with the specified target group.
remove_listener_certificates
Remove the specified certificate from the certificate list for the specified HTTPS or TLS listener.
remove_tags
Remove the specified tags from the specified Elastic Load Balancing resources.
remove_trust_store_revolutions
Remove the specified revocation file from the specified trust store.
set_ip_address_type
Sets the type of IP addresses used by the subnets of the specified load balancer.
set_rule_priorities
Sets the priorities of the specified rules.
set_security_groups
Associate the specified security groups with the specified Application Load Balancer or Network Load Balancer.
set_subnets
Enable the Availability Zones for the specified public subnets for the specified Application Load Balancer, Network Load Balancer, or Gateway Load Balancer.

Examples

```r
## Not run:
svc <- elbv2()
# This example adds the specified tags to the specified load balancer.
```
emr = library(emr)

emr$add_tags(
  ResourceArns = list(
    "arn:aws:elasticloadbalancing:us-west-2:123456789012:loadbalancer/app/m..."
  ),
  Tags = list(
    list(
      Key = "project",
      Value = "lima"
    ),
    list(
      Key = "department",
      Value = "digital-media"
    )
  )
)

## End(Not run)

---

**Amazon EMR**

**Description**

Amazon EMR is a web service that makes it easier to process large amounts of data efficiently. Amazon EMR uses Hadoop processing combined with several Amazon Web Services services to do tasks such as web indexing, data mining, log file analysis, machine learning, scientific simulation, and data warehouse management.

**Usage**

`emr(config = list(), credentials = list(), endpoint = NULL, region = NULL)`

**Arguments**

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

• **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoint.html

**credentials** Optional credentials shorthand for the config parameter

• **creds**:  
  – **access_key_id**: AWS access key ID  
  – **secret_access_key**: AWS secret access key  
  – **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

### Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

### Service syntax

```r
svc <- emr(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  )
)
```
secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

add_instance_fleet Adds an instance fleet to a running cluster
add_instance_groups Adds one or more instance groups to a running cluster
add_job_flow_steps AddJobFlowSteps adds new steps to a running cluster
cancel_steps Cancels a pending step or steps in a running cluster
create_security_configuration Creates a security configuration, which is stored in the service and can be specified when a cluster is created
create_studio Creates a new Amazon EMR Studio
create_studio_session_mapping Maps a user or group to the Amazon EMR Studio specified by StudioId, and applies a security configuration
    Delete a security configuration
    delete_security_configuration
    delete_studio Removes an Amazon EMR Studio from the Studio metadata store
    delete_studio_session_mapping Removes a user or group from an Amazon EMR Studio
    describe_cluster Provides cluster-level details including status, hardware and software configuration, VPC settings, and so on
    describe_job_flows This API is no longer supported and will eventually be removed
    describe_notebook_execution Provides details of a notebook execution
    describe_release_label Provides Amazon EMR release label details, such as the releases available the Region where the API request is run, and the available applications for a specific Amazon EMR release label
    describe_security_configuration Provides the details of a security configuration by returning the configuration JSON
    describe_step Provides more detail about the cluster step
    describe_studio Returns details for the specified Amazon EMR Studio including ID, Name, VPC, Studio access URL, and so on
    get_auto_termination_policy Returns the auto-termination policy for an Amazon EMR cluster
    get_block_public_access_configuration Returns the Amazon EMR block public access configuration for your Amazon Web Services account
    get_cluster_session_credentials Provides temporary, HTTP basic credentials that are associated with a given runtime IAM role
    get_managed_scaling_policy Fetches the attached managed scaling policy for an Amazon EMR cluster
    get_studio_session_mapping Fetches mapping details for the specified Amazon EMR Studio and identity (user or group)
    list_bootstrap_actions Provides information about the bootstrap actions associated with a cluster
    list_clusters Provides the status of all clusters visible to this Amazon Web Services account
    list_instance_fleets Lists all available details about the instance fleets in a cluster
    list_instance_groups Lists all available details about the instance groups in a cluster
    list_instances Provides information for all active Amazon EC2 instances and Amazon EC2 instance-related metadata
    list_notebook_executions Provides summaries of all notebook executions
    list_release_labels Provides release labels of Amazon EMR services in the Region where the API is called
    list_security_configurations Lists all the security configurations visible to this account, providing their creation dates and times, and their names
    list_steps Provides a list of steps for the cluster in reverse order unless you specify stepIds with the request or filter by StepStates
    list_studios Returns a list of all Amazon EMR Studios associated with the Amazon Web Services account
    list_studio_session_mappings Returns a list of all user or group session mappings for the Amazon EMR Studio specified by StudioId
    list_supported_instance_types A list of the instance types that Amazon EMR supports
    modify_cluster Modifies the number of steps that can be executed concurrently for the cluster specified using ClusterId
modify_instance_fleet
modify_instance_groups
put_auto_scaling_policy
put_auto_termination_policy
put_block_public_access_configuration
put_managed_scaling_policy
remove_auto_scaling_policy
remove_auto_termination_policy
remove_managed_scaling_policy
remove_tags
run_job_flow
set_keep_job_flow_alive_when_no_steps
set_termination_protection
set_unhealthy_node_replacement
set_visible_to_all_users
start_notebook_execution
stop_notebook_execution
terminate_job_flows
update_studio
update_studio_session_mapping

Modifies the target On-Demand and target Spot capacities for the instance fleet with the specified InstanceFleetID within the cluster specified using ClusterID. ModifyInstanceGroups modifies the number of nodes and configuration settings of an instance group. Auto-termination is supported in Amazon EMR releases 5 and later. Creates or updates an automatic scaling policy for a core instance group or task instance group in an Amazon EMR cluster. Creates or updates a managed scaling policy for an Amazon EMR cluster. Removes an automatic scaling policy from a specified instance group within an Amazon EMR cluster. Removes an auto-termination policy from an Amazon EMR cluster. Removes a managed scaling policy from a specified Amazon EMR cluster. Removes tags from an Amazon EMR resource, such as a cluster or Amazon EMR Studio. RunJobFlow creates and starts running a new cluster (job flow). You can use the SetKeepJobFlow AliveWhenNoSteps to configure a cluster (job flow) so that it does not terminate when all its steps complete. SetTerminationProtection locks a cluster (job flow) so the Amazon EC2 instances in the cluster cannot be terminated by user intervention, an API call, or in the event of a job-flow error. SetUnhealthyNodeReplacement lets Amazon EMR gracefully replace core nodes on a cluster if any nodes become unhealthy. The SetVisibleToAllUsers parameter is no longer supported. Starts a notebook execution. Stops a notebook execution. TerminateJobFlows shuts a list of clusters (job flows) down. Updates an Amazon EMR Studio configuration, including attributes such as name, description, and subnets. Updates the session policy attached to the user or group for the specified Amazon EMR Studio.

Examples

```r
## Not run:
svc <- emr()
svc$add_instance_fleet(
  Foo = 123
)

## End(Not run)
```

Amazon EMR on EKS provides a deployment option for Amazon EMR that allows you to run open-source big data frameworks on Amazon Elastic Kubernetes Service (Amazon EKS). With this deployment option, you can focus on running analytics workloads while Amazon EMR on EKS builds, configures, and manages containers for open-source applications. For more information about Amazon EMR on EKS concepts and tasks, see What is Amazon EMR on EKS.

Amazon EMR containers is the API name for Amazon EMR on EKS. The emr-containers prefix is used in the following scenarios:

- It is the prefix in the CLI commands for Amazon EMR on EKS. For example, aws emr-containers start-job-run.
- It is the prefix before IAM policy actions for Amazon EMR on EKS. For example, "Action": [ "emr-containers:StartJobRun"]. For more information, see Policy actions for Amazon EMR on EKS.
- It is the prefix used in Amazon EMR on EKS service endpoints. For example, emr-containers.us-east-2.amazonaws.com. For more information, see Amazon EMR on EKSService Endpoints.

Usage

```r
emrcontainers(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL)
```

Arguments

- **config** Optional configuration of credentials, endpoint, and/or region.
  - **credentials**: • creds:
    - access_key_id: AWS access key ID
    - secret_access_key: AWS secret access key
    - session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.
  - endpoint: The complete URL to use for the constructed client.
  - region: The AWS Region used in instantiating the client.
  - close_connection: Immediately close all HTTP connections.
  - timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  - sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html

- **credentials** Optional credentials shorthand for the config parameter
  - creds:
    - access_key_id: AWS access key ID
    - secret_access_key: AWS secret access key
    - session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.

- **endpoint** Optional shorthand for complete URL to use for the constructed client.

- **region** Optional shorthand for AWS Region used in instantiating the client.
Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- emrcontainers(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string"
  )
)
```

Operations

- `cancel_job_run`: Cancels a job run
- `create_job_template`: Creates a job template
- `create_managed_endpoint`: Creates a managed endpoint
- `create_security_configuration`: Creates a security configuration
- `create_virtual_cluster`: Creates a virtual cluster
- `delete_job_template`: Deletes a job template
- `delete_managed_endpoint`: Deletes a managed endpoint
- `delete_virtual_cluster`: Deletes a virtual cluster
describe_job_run
describe_job_template
describe_managed_endpoint
describe_security_configuration
describe_virtual_cluster
get_managed_endpoint_session_credentials
list_job_runs
list_job_templates
list.managed_endpoints
list_security_configurations
list_tags_for_resource
list_virtual_clusters
start_job_run
tag_resource
untag_resource

Displays detailed information about a job run
Displays detailed information about a specified job template
Displays detailed information about a managed endpoint
Displays detailed information about a specified security configuration
Displays detailed information about a specified virtual cluster
Generate a session token to connect to a managed endpoint
Lists job runs based on a set of parameters
Lists job templates based on a set of parameters
Lists managed endpoints based on a set of parameters
Lists security configurations based on a set of parameters
Lists the tags assigned to the resources
Lists information about the specified virtual cluster
Starts a job run
Assigns tags to resources
Removes tags from resources

Examples

```r
## Not run:
svc <- emrcontainers()
svc$cancel_job_run(
  Foo = 123
)

## End(Not run)
```

emrserverless

**EMR Serverless**

Description

Amazon EMR Serverless is a new deployment option for Amazon EMR. Amazon EMR Serverless provides a serverless runtime environment that simplifies running analytics applications using the latest open source frameworks such as Apache Spark and Apache Hive. With Amazon EMR Serverless, you don’t have to configure, optimize, secure, or operate clusters to run applications with these frameworks.

The API reference to Amazon EMR Serverless is **emr-serverless**. The **emr-serverless** prefix is used in the following scenarios:

- It is the prefix in the CLI commands for Amazon EMR Serverless. For example, `aws emr-serverless start-job-run`.
- It is the prefix before IAM policy actions for Amazon EMR Serverless. For example, "Action": ["emr-serverless:StartJobRun"]. For more information, see **Policy actions for Amazon EMR Serverless**.
- It is the prefix used in Amazon EMR Serverless service endpoints. For example, `emr-serverless.us-east-2.amazonaws.com`.

For more information, see the [Amazon EMR Serverless documentation](https://docs.aws.amazon.com/emr-serverless/latest/devguide/emr-serverless.html).
Usage

```r
emrserverless(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **creds**:
    - *access_key_id*: AWS access key ID
    - *secret_access_key*: AWS secret access key
    - *session_token*: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.
- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - *access_key_id*: AWS access key ID
    - *secret_access_key*: AWS secret access key
    - *session_token*: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.
- **endpoint**: Optional shorthand for complete URL to use for the constructed client.
- **region**: Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```r
svc <- emrserverless(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `cancel_job_run`: Cancels a job run
- `create_application`: Creates an application
- `delete_application`: Deletes an application
- `get_application`: Displays detailed information about a specified application
- `get_dashboard_for_job_run`: Creates and returns a URL that you can use to access the application UIs for a job run
- `get_job_run`: Displays detailed information about a job run
- `list_applications`: Lists applications based on a set of parameters
- `list_job_runs`: Lists job runs based on a set of parameters
- `list_tags_for_resource`: Lists the tags assigned to the resources
- `start_application`: Starts a specified application and initializes initial capacity if configured
- `start_job_run`: Starts a job run
- `stop_application`: Stops a specified application and releases initial capacity if configured
- `tag_resource`: Assigns tags to resources
- `untag_resource`: Removes tags from resources
entityresolution

update_application

Updates a specified application

Examples

```r
## Not run:
svc <- emrserverless()
svc$cancel_job_run(
  Foo = 123
)

## End(Not run)
```

description

Welcome to the Entity Resolution API Reference.

Entity Resolution is an Amazon Web Services service that provides pre-configured entity resolution capabilities that enable developers and analysts at advertising and marketing companies to build an accurate and complete view of their consumers.

With Entity Resolution, you can match source records containing consumer identifiers, such as name, email address, and phone number. This is true even when these records have incomplete or conflicting identifiers. For example, Entity Resolution can effectively match a source record from a customer relationship management (CRM) system with a source record from a marketing system containing campaign information.

To learn more about Entity Resolution concepts, procedures, and best practices, see the Entity Resolution User Guide.

Usage

```r
description

entityresolution(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- **config**
  Optional configuration of credentials, endpoint, and/or region.

  - **credentials**:
    - **creds**:

```r
description
```
* **access_key_id**: AWS access key ID
* **secret_access_key**: AWS secret access key
* **session_token**: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.

- **anonymous**: Set anonymous credentials.

- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials**

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.

- **anonymous**: Set anonymous credentials.

**endpoint**

- **endpoint**: Optional shorthand for complete URL to use for the constructed client.

**region**

- **region**: Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- entityresolution(
    config = list(
        credentials = list(
            creds = list(
                access_key_id = "string",
                secret_access_key = "string",
                session_token = "string"
            ),
            profile = "string",
            anonymous = "logical"
        ),
```
),
endpoint = "string",
region = "string",
close_connection = "logical",
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

add_policy_statement  Adds a policy statement object
batch_delete_unique_id  Deletes multiple unique IDs in a matching workflow
create_id_mapping_workflow  Creates an IdMappingWorkflow object which stores the configuration of the data processing job
create_id_namespace  Creates an ID namespace object which will help customers provide metadata explaining their dataset and how to use it
create_matching_workflow  Creates a MatchingWorkflow object which stores the configuration of the data processing job
create_schema_mapping  Creates a schema mapping, which defines the schema of the input customer records table
delete_id_mapping_workflow  Deletes the IdMappingWorkflow with a given name
delete_id_namespace  Deletes the IdNamespace with a given name
delete_matching_workflow  Deletes the MatchingWorkflow with a given name
delete_policy_statement  Deletes the policy statement
delete_schema_mapping  Deletes the SchemaMapping with a given name
get_id_mapping_job  Gets the status, metrics, and errors (if there are any) that are associated with a job
get_id_mapping_workflow  Returns the IdMappingWorkflow with a given name, if it exists
get_id_namespace  Returns the IdNamespace with a given name, if it exists
get_match_id  Returns the corresponding Match ID of a customer record if the record has been processed
get_matching_job  Gets the status, metrics, and errors (if there are any) that are associated with a job
get_matching_workflow  Returns the MatchingWorkflow with a given name, if it exists
get_policy  Returns the resource-based policy
get_provider_service  Returns the ProviderService of a given name
get_schema_mapping  Returns the SchemaMapping of a given name
list_id_mapping_jobs  Lists all ID mapping jobs for a given workflow
list_id_mapping_workflows  Returns a list of all the IdMappingWorkflows that have been created for an Amazon Web Services account
list_id_namespaces  Returns a list of all ID namespaces
list_matching_jobs  Lists all jobs for a given workflow
list_matching_workflows  Returns a list of all the MatchingWorkflows that have been created for an Amazon Web Services account
### Examples

```r
## Not run:
svc <- entityresolution()
svc$add_policy_statement(
  Foo = 123
)

## End(Not run)
```

---

### Description

Amazon EventBridge helps you to respond to state changes in your Amazon Web Services resources. When your resources change state, they automatically send events to an event stream. You can create rules that match selected events in the stream and route them to targets to take action. You can also use rules to take action on a predetermined schedule. For example, you can configure rules to:

- Automatically invoke an Lambda function to update DNS entries when an event notifies you that Amazon EC2 instance enters the running state.
- Direct specific API records from CloudTrail to an Amazon Kinesis data stream for detailed analysis of potential security or availability risks.
- Periodically invoke a built-in target to create a snapshot of an Amazon EBS volume.

For more information about the features of Amazon EventBridge, see the [Amazon EventBridge User Guide](#).
eventbridge

Usage

```r
eventbridge(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **creds**:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - `profile`: The name of a profile to use. If not given, then the default profile is used.
  - `anonymous`: Set anonymous credentials.
- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [documentation](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - `creds`:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - `profile`: The name of a profile to use. If not given, then the default profile is used.
  - `anonymous`: Set anonymous credentials.
- **endpoint**: Optional shorthand for complete URL to use for the constructed client.
- **region**: Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```
svc <- eventbridge(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- **activate_event_source**: Activates a partner event source that has been deactivated
- **cancel_replay**: Cancels the specified replay
- **create_api_destination**: Creates an API destination, which is an HTTP invocation endpoint configured as a target for events
- **create_archive**: Creates an archive of events with the specified settings
- **create_connection**: Creates a connection
- **create_endpoint**: Creates a global endpoint
- **create_event_bus**: Creates a new event bus within your account
- **create_partner_event_source**: Called by an SaaS partner to create a partner event source
- **deactivate_event_source**: You can use this operation to temporarily stop receiving events from the specified partner event source
- **deauthorize_connection**: Removes all authorization parameters from the connection
- **delete_api_destination**: Deletes the specified API destination
- **delete_archive**: Deletes the specified archive
- **delete_connection**: Deletes a connection
- **delete_endpoint**: Delete an existing global endpoint
delete_event_bus
delete_partner_event_source
delete_rule
describe_api_destination
describe_archive
describe_connection
describe_endpoint
describe_event_bus
describe_event_source
describe_partner_event_source
describe_replay
describe_rule
disable_rule
enable_rule
list_api_destinations
list_archives
list_connections
list_endpoints
list_event_buses
list_event_sources
list_partner_event_source_accounts
list_partner_event_sources
list_replays
list_rule_names_by_target
list_rules
list_tags_for_resource
list_targets_by_rule
put_events
put_partner_events
put_permission
put_rule
put_targets
remove_permission
remove_targets
start_replay
tag_resource
test_event_pattern
untag_resource
update_api_destination
update_archive
update_connection
update_endpoint

Examples
## Not run:
svc <- eventbridge()
eventbridgepipes

## Description

Amazon EventBridge Pipes connects event sources to targets. Pipes reduces the need for specialized knowledge and integration code when developing event driven architectures. This helps ensure consistency across your company’s applications. With Pipes, the target can be any available EventBridge target. To set up a pipe, you select the event source, add optional event filtering, define optional enrichment, and select the target for the event data.

## Usage

```r
eventbridgepipes(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

## Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.

  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.

  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.

```r
events$activate_event_source(
  Foo = 123
)
```

## End (Not run)
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials**

Optional credentials shorthand for the config parameter

• **creds**:
  
  – **access_key_id**: AWS access key ID
  
  – **secret_access_key**: AWS secret access key
  
  – **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

**endpoint**

Optional shorthand for complete URL to use for the constructed client.

**region**

Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- eventbridgepipes(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
  
```

eventbridgescheduler

## Anonymous

```r
anonymous = "logical"
```

## Endpoint

```r
endpoint = "string",
region = "string"
```

### Operations

- **create_pipe** Create a pipe
- **delete_pipe** Delete an existing pipe
- **describe_pipe** Get the information about an existing pipe
- **list_pipes** Get the pipes associated with this account
- **list_tags_for_resource** Displays the tags associated with a pipe
- **start_pipe** Start an existing pipe
- **stop_pipe** Stop an existing pipe
- **tag_resource** Assigns one or more tags (key-value pairs) to the specified pipe
- **untag_resource** Removes one or more tags from the specified pipes
- **update_pipe** Update an existing pipe

### Examples

```r
## Not run:
svc <- eventbridgescheduler()
svc$create_pipe(
  Foo = 123
)
## End(Not run)
```

---

**eventbridgescheduler** *Amazon EventBridge Scheduler*

### Description

Amazon EventBridge Scheduler is a serverless scheduler that allows you to create, run, and manage tasks from one central, managed service. EventBridge Scheduler delivers your tasks reliably, with built-in mechanisms that adjust your schedules based on the availability of downstream targets. The following reference lists the available API actions, and data types for EventBridge Scheduler.

### Usage

```r
eventbridgescheduler(
  config = list(),
  credentials = list(),
```
Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**: The complete URL to use for the constructed client.
  - **endpoint**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

- **endpoint**: Optional shorthand for complete URL to use for the constructed client.

- **region**: Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...),` where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```r
svc <- eventbridgescheduler(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `create_schedule`: Creates the specified schedule
- `create_schedule_group`: Creates the specified schedule group
- `delete_schedule`: Deletes the specified schedule
- `delete_schedule_group`: Deletes the specified schedule group
- `get_schedule`: Retrieves the specified schedule
- `get_schedule_group`: Retrieves the specified schedule group
- `list_schedule_groups`: Returns a paginated list of your schedule groups
- `list_schedules`: Returns a paginated list of your EventBridge Scheduler schedules
- `list_tags_for_resource`: Lists the tags associated with the Scheduler resource
- `tag_resource`: Assigns one or more tags (key-value pairs) to the specified EventBridge Scheduler resource
- `untag_resource`: Removes one or more tags from the specified EventBridge Scheduler schedule group
- `update_schedule`: Updates the specified schedule
Examples

```r
## Not run:
svc <- eventbridgescheduler()
svc$create_schedule(
  Foo = 123
)
## End(Not run)
```

### Description

The FinSpace management service provides the APIs for managing FinSpace environments.

### Usage

```r
finspace(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

### Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**: Optional credentials shorthand for the config parameter
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.

- **credentials**: Optional credentials shorthand for the config parameter
• **creds:**
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
• **profile**: The name of a profile to use. If not given, then the default profile is used.
• **anonymous**: Set anonymous credentials.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- finspace(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```
Operations

create_environment
create_kx_changeset
create_kx_cluster
create_kx_database
create_kx_dataview
create_kx_environment
create_kx_scaling_group
create_kx_user
create_kx_volume
delete_environment
delete_kx_cluster
delete_kx_cluster_node
delete_kx_database
delete_kx_dataview
delete_kx_environment
delete_kx_scaling_group
delete_kx_user
delete_kx_volume
get_environment
get_kx_changeset
get_kx_cluster
get_kx_connection_string
get_kx_database
get_kx_dataview
get_kx_environment
get_kx_scaling_group
get_kx_user
get_kx_volume
list_environments
list_kx_changesets
list_kx_cluster_nodes
list_kx_clusters
list_kx_databases
list_kx_dataviews
list_kx_environments
list_kx_scaling_groups
list_kx_users
list_kx_volumes
list_tags_for_resource
tag_resource
untag_resource
update_environment
update_kx_cluster_code_configuration
update_kx_cluster_databases
update_kx_database
update_kx_dataview

Create a new FinSpace environment
Creates a changeset for a kdb database
Creates a new kdb cluster
Creates a new kdb database in the environment
Creates a snapshot of kdb database with tiered storage capabilities and a pre-warmed cache
Creates a managed kdb environment for the account
Creates a new scaling group
Creates a user in FinSpace kdb environment with an associated IAM role
Creates a new volume with a specific amount of throughput and storage capacity
Delete an FinSpace environment
Deletes a kdb cluster
Deletes the specified nodes from a cluster
Deletes the specified database and all of its associated data
Deletes the specified dataview
Deletes the kdb environment
Deletes the specified scaling group
Deletes a user in the specified kdb environment
Deletes a volume
Returns the FinSpace environment object
Returns information about a kdb changeset
Returns information about a kdb cluster
Retrieves a connection string for a user to connect to a kdb cluster
Returns database information for the specified environment ID
Retrieves all the information for the specified kdb environment
Retrieves details of a scaling group
Retrieves information about the specified kdb user
Retrieves the information about the volume
A list of all of your FinSpace environments
Returns a list of all the changesets for a database
Lists all the nodes in a kdb cluster
Returns a list of clusters
Returns a list of all the databases in the kdb environment
Returns a list of all the dataviews in the database
Returns a list of kdb environments created in an account
Returns a list of scaling groups in a kdb environment
Lists all the users in a kdb environment
Lists all the volumes in a kdb environment
A list of all tags for a resource
Adds metadata tags to a FinSpace resource
Removes metadata tags from a FinSpace resource
Update your FinSpace environment
Allows you to update code configuration on a running cluster
Updates the databases mounted on a kdb cluster, which includes the changesetId and all the dbPaths to be cached
Updates information for the given kdb database
Updates the specified dataview
update_kx_environment  Updates information for the given kdb environment
update_kx_environment_network  Updates environment network to connect to your internal network by using a transit gateway
update_kx_user  Updates the user details
update_kx_volume  Updates the throughput or capacity of a volume

Examples

```r
## Not run:
svc <- finspace()
svc$create_environment(
  Foo = 123
)
## End(Not run)
```

---

**finspacedata**  
*FinSpace Public API*

**Description**

The FinSpace APIs let you take actions inside the FinSpace.

**Usage**

```r
finspacedata(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

**Arguments**

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
• **region**: The AWS Region used in instantiating the client.
• **close_connection**: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

### credentials
Optional credentials shorthand for the config parameter

• **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
• **profile**: The name of a profile to use. If not given, then the default profile is used.
• **anonymous**: Set anonymous credentials.

### endpoint
Optional shorthand for complete URL to use for the constructed client.

### region
Optional shorthand for AWS Region used in instantiating the client.

### Value
A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

### Service syntax

```r
svc <- finspacedata(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  
```
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

associate_user_to_permission_group
create_changeset
create_dataset
create_data_view
create_permission_group
create_user
delete_dataset
delete_permission_group
disable_user
disassociate_user_from_permission_group
enable_user
get_changeset
get_dataset
get_data_view
get_external_data_view_access_details
get_permission_group
get_programmatic_access_credentials
get_user
get_working_location
list_changesets
list_datasets
list_data_views
list_permission_groups
list_permission_groups_by_user
list_users
list_users_by_permission_group
reset_user_password
update_changeset
update_dataset
update_permission_group
update_user

Adds a user to a permission group to grant permissions for actions a user can perform in FinSpace
Creates a new Changeset in a FinSpace Dataset
Creates a new FinSpace Dataset
Creates a Dataview for a Dataset
Creates a group of permissions for various actions that a user can perform in FinSpace
Creates a new user in FinSpace
Deletes a FinSpace Dataset
Deletes a permission group
Denies access to the FinSpace web application and API for the specified user
Removes a user from a permission group
Allows the specified user to access the FinSpace web application and API
Gets information about a Changeset
Returns information about a Dataset
Gets information about a Dataview
Returns the credentials to access the external Dataview from an S3 location
Retrieves the details of a specific permission group
Request programmatic credentials to use with FinSpace SDK
Retrieves details for a specific user
A temporary Amazon S3 location, where you can copy your files from a source location
Lists the FinSpace Changesets for a Dataset
Lists all of the active Datasets that a user has access to
Lists all available Dataviews for a Dataset
Lists all available permission groups in FinSpace
Lists all the permission groups that are associated with a specific user
Lists all available users in FinSpace
Lists details of all the users in a specific permission group
Resets the password for a specified user ID and generates a temporary one
Updates a FinSpace Changeset
Updates a FinSpace Dataset
Modifies the details of a permission group
Modifies the details of the specified user
### Description

Amazon Data Firehose

Amazon Data Firehose was previously known as Amazon Kinesis Data Firehose.

Amazon Data Firehose is a fully managed service that delivers real-time streaming data to destinations such as Amazon Simple Storage Service (Amazon S3), Amazon OpenSearch Service, Amazon Redshift, Splunk, and various other supported destinations.

### Usage

```r
firehose(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

### Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      * **access_key_id**: AWS access key ID
      * **secret_access_key**: AWS secret access key
      * **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

credentials  Optional credentials shorthand for the config parameter

• **creds**:  
  – **access_key_id**: AWS access key ID
  – **secret_access_key**: AWS secret access key
  – **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

endpoint  Optional shorthand for complete URL to use for the constructed client.

region  Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...),` where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- firehose(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
```
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

create_delivery_stream  Creates a Firehose delivery stream
delete_delivery_stream Deletes a delivery stream and its data
describe_delivery_stream Describes the specified delivery stream and its status
list_delivery_streams Lists your delivery streams in alphabetical order of their names
list_tags_for_delivery_stream Lists the tags for the specified delivery stream
put_record Writes a single data record into an Amazon Firehose delivery stream
put_record_batch Writes multiple data records into a delivery stream in a single call, which can achieve high
start_delivery_stream_encryption Enables server-side encryption (SSE) for the delivery stream
stop_delivery_stream_encryption Disables server-side encryption (SSE) for the delivery stream
tag_delivery_stream Adds or updates tags for the specified delivery stream
untag_delivery_stream Removes tags from the specified delivery stream
update_destination Updates the specified destination of the specified delivery stream

Examples

## Not run:
svc <- firehose()
svc$create_delivery_stream(
  Foo = 123
)

## End(Not run)

---

**fis**  
*AWS Fault Injection Simulator*

**Description**

Fault Injection Service is a managed service that enables you to perform fault injection experiments on your Amazon Web Services workloads. For more information, see the Fault Injection Service User Guide.

**Usage**

fis(config = list(), credentials = list(), endpoint = NULL, region = NULL)
Arguments

config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

directional Optional shorthand for complete URL to use for the constructed client.
region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(…), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

svc <- fis(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
    )
  )
)
Operations

create_experiment_template
create_target_account_configuration
delete_experiment_template
delete_target_account_configuration
get_action
get_experiment
get_experiment_target_account_configuration
get_experiment_template
get_target_account_configuration
get_target_resource_type
list_actions
list_experiment_resolved_targets
list_experiments
list_experiment_target_account_configurations
list_experiment_templates
list_tags_for_resource
list_target_account_configurations
list_target_resource_types
start_experiment
stop_experiment

Creates an experiment template
Creates a target account configuration for the experiment template
Deletes the specified experiment template
Deletes the specified target account configuration of the experiment template
Gets information about the specified FIS action
Gets information about the specified experiment
Gets information about the specified target account configuration of the experiment
Gets information about the specified experiment template
Gets information about the specified target account configuration of the experiment
Gets information about the specified resource type
Lists the available FIS actions
Lists the resolved targets information of the specified experiment
Lists your experiments
Lists the target account configurations of the specified experiment
Lists your experiment templates
Lists the tags for the specified resource
Lists the target account configurations of the specified experiment template
Lists the target resource types
Starts running an experiment from the specified experiment template
Stops the specified experiment
Examples

```r
## Not run:
svc <- fis()
svc$create_experiment_template(
  Foo = 123
)
## End(Not run)
```

**fms**

*Firewall Management Service*

**Description**

This is the *Firewall Manager API Reference*. This guide is for developers who need detailed information about the Firewall Manager API actions, data types, and errors. For detailed information about Firewall Manager features, see the *Firewall Manager Developer Guide*.

Some API actions require explicit resource permissions. For information, see the developer guide topic *Service roles for Firewall Manager*.

**Usage**

```r
fms(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

**Arguments**

<table>
<thead>
<tr>
<th>config</th>
<th>Optional configuration of credentials, endpoint, and/or region.</th>
</tr>
</thead>
<tbody>
<tr>
<td>credentials</td>
<td></td>
</tr>
<tr>
<td>cred</td>
<td></td>
</tr>
<tr>
<td>access_key_id</td>
<td>AWS access key ID</td>
</tr>
<tr>
<td>secret_access_key</td>
<td>AWS secret access key</td>
</tr>
<tr>
<td>session_token</td>
<td>AWS temporary session token</td>
</tr>
<tr>
<td>profile</td>
<td>The name of a profile to use. If not given, then the default profile is used.</td>
</tr>
<tr>
<td>anonymous</td>
<td>Set anonymous credentials.</td>
</tr>
<tr>
<td>endpoint</td>
<td>The complete URL to use for the constructed client.</td>
</tr>
<tr>
<td>region</td>
<td>The AWS Region used in instantiating the client.</td>
</tr>
</tbody>
</table>
• **close_connection**: Immediately close all HTTP connections.

• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

• **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

credentials Optional credentials shorthand for the config parameter

• **creds**:  
  – **access_key_id**: AWS access key ID  
  – **secret_access_key**: AWS secret access key  
  – **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- fms(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
```
creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

associate_admin_account
associate_third_party_firewall
batch_associate_resource
batch_disassociate_resource
delete_apps_list
delete_notification_channel
delete_policy
delete_protocols_list
delete_resource_set
disassociate_admin_account
disassociate_third_party_firewall
get_admin_account
get_admin_scope
get_apps_list
get_compliance_detail
get_notification_channel
get_policy
get_protection_status
get_protocols_list
get_resource_set
get_third_party_firewall_association_status
getViolationDetails
list_admin_accounts_for_organization
list_admins_managing_account
list_apps_lists
list_compliance_status
list_discovered_resources
list_member_accounts
list_policies
list_protocols_lists
list_resource_set_resources
list_resource_sets
list_tags_for_resource
list_third_party_firewall_firewall_policies

Sets a Firewall Manager default administrator account
Sets the Firewall Manager policy administrator as a tenant administrator of a third-party firewall vendor
Associate resources to a Firewall Manager resource set
Disassociates resources from a Firewall Manager resource set
Permanently deletes a Firewall Manager applications list
Deletes an Firewall Manager association with the IAM role and the Amazon Simple Notification Service (SNS) topic that is used to record Firewall Manager SNS logs
Permanently deletes an Firewall Manager policy
Permanently deletes an Firewall Manager protocols list
Deletes the specified ResourceSet
Disassociates an Firewall Manager administrator account
Disassociates a Firewall Manager policy administrator from a third-party firewall vendor
Returns the Organizations account that is associated with Firewall Manager as the Firewall Manager default administrator
Returns information about the specified account’s administrative scope
Returns information about the specified Firewall Manager applications list
Returns detailed compliance information about the specified member account
Information about the Amazon Simple Notification Service (SNS) topic that is used to record Firewall Manager SNS logs
Returns information about the specified Firewall Manager policy
If you created a Shield Advanced policy, returns policy-level attack summary information in the event of a potential DDoS attack
Returns information about the specified Firewall Manager protocols list
Gets information about a specific resource set
The onboarding status of a Firewall Manager admin account to third-party firewall vendor
Retrieves violations for a resource based on the specified Firewall Manager policy
Returns a AdminAccounts object that lists the Firewall Manager administrators of the specified account
Lists the accounts that are managing the specified Organizations member account
Returns an array of AppsListDataSummary objects
Returns an array of PolicyComplianceStatus objects
Returns an array of resources in the organization’s accounts that are available to the account
Returns a MemberAccounts object that lists the member accounts in the administrator account
Returns an array of PolicySummary objects
Returns an array of ProtocolsListDataSummary objects
Returns an array of resources that are currently associated to a resource set
Returns an array of ResourceSetSummary objects
Retrieves the list of tags for the specified Amazon Web Services resource
Retrieves a list of all of the third-party firewall policies that are associated with the specified account
### forecastqueryservice

**Description**

Provides APIs for creating and managing Amazon Forecast resources.

**Usage**

```r
forecastqueryservice(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

**Arguments**

- `config`: Optional configuration of credentials, endpoint, and/or region.
  - `credentials`: Optional configuration of credentials, endpoint, and/or region.
    - `creds`:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

### credentials
Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

### endpoint
Optional shorthand for complete URL to use for the constructed client.

### region
Optional shorthand for AWS Region used in instantiating the client.

### Value
A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

### Service syntax
```
svc <- forecastqueryservice(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
  
```
Operations

query_forecast Retrieves a forecast for a single item, filtered by the supplied criteria
query_what_if_forecast Retrieves a what-if forecast

Examples

```r
## Not run:
svc <- forecastqueryservice()
svc$query_forecast(
   Foo = 123
)
## End(Not run)
```

Description

Provides APIs for creating and managing Amazon Forecast resources.

Usage

```r
forecastservice(
   config = list(),
   credentials = list(
      creds = list(
         access_key_id = "string",
         secret_access_key = "string",
         session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
   ),
   endpoint = "string",
   region = "string"
)
```
endpoint = NULL,
        region = NULL
    )

Arguments

config Optional configuration of credentials, endpoint, and/or region.

  • credentials:
    – creds:
      • access_key_id: AWS access key ID
      • secret_access_key: AWS secret access key
      • session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.

  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_forced_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends html

credentials Optional credentials shorthand for the config parameter

  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```r
svc <- forecastservice(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `create_auto_predictor`: Creates an Amazon Forecast predictor
- `create_dataset`: Creates an Amazon Forecast dataset
- `create_dataset_group`: Creates a dataset group, which holds a collection of related datasets
- `create_dataset_import_job`: Imports your training data to an Amazon Forecast dataset
- `create_explainability`: Explainability is only available for Forecasts and Predictors generated from an AutoPredictor. It creates an Explainability resource created by the CreateExplainability operation
- `create_explainability_export`: Exports an Explainability resource created by the CreateExplainability operation
- `create_forecast`: Creates a forecast for each item in the TARGET_TIME_SERIES dataset that was used to train the predictor
- `create_forecast_export_job`: Exports a forecast created by the CreateForecast operation to your Amazon Simple Storage Service (Amazon S3) bucket
- `create_monitor`: Creates a predictor monitor resource for an existing auto predictor
- `create_predictor`: This operation creates a legacy predictor that does not include all the predictor functionalities provided by Amazon Forecast
- `create_predictor_backtest_export_job`: Exports backtest forecasts and accuracy metrics generated by the CreateAutoPredictor operation
- `create_what_if_analysis`: What-if analysis is a scenario modeling technique where you make a hypothetical change to a time series and compare the forecasts generated by these changes against the baseline, unchanged time series
- `create_what_if_forecast`: A what-if forecast is a forecast that is created from a modified version of the baseline time series
- `create_what_if_forecast_export`: Exports a forecast created by the CreateWhatIfForecast operation to your Amazon Simple Storage Service (Amazon S3) bucket
delete_dataset
delete_dataset_group
delete_dataset_import_job
delete_explainability
delete_explainability_export
delete_forecast
delete_forecast_export_job
delete_monitor
delete_predictor
delete_predictor_backtest_export_job
delete_resource_tree
delete_what_if_analysis
delete_what_if_forecast
describe_auto_predictor
describe_dataset
describe_dataset_group
describe_dataset_import_job
describe_explainability
describe_explainability_export
describe_forecast
describe_forecast_export_job
describe_monitor
describe_predictor
describe_predictor_backtest_export_job
describe_what_if_analysis
describe_what_if_forecast
describe_what_if_forecast_export
get_accuracy_metrics
list_dataset_groups
list_dataset_import_jobs
list_datasets
list_explainabilities
list_explainability_exports
list_forecast_export_jobs
list_forecasts
list_monitor_evaluations
list_monitors
list_predictor_backtest_export_jobs
list_predictors
list_tags_for_resource
list_what_if_analyses
list_what_if_forecast_exports
list_what_if_forecasts
resume_resource
stop_resource
tag_resource
untag_resource

Deletes an Amazon Forecast dataset that was created using the CreateDataset operation.
Deletes a dataset group created using the CreateDatasetGroup operation.
Deletes a dataset import job created using the CreateDatasetImportJob operation.
Deletes an Explainability resource.
Deletes an Explainability export.
Deletes a forecast created using the CreateForecast operation.
Deletes a forecast export job created using the CreateForecastExportJob operation.
Deletes a monitor resource.
Deletes a predictor created using the DescribePredictor or CreatePredictor operation.
Deletes a predictor backtest export job.
Deletes an entire resource tree.
Deletes a what-if analysis created using the CreateWhatIfAnalysis operation.
Deletes a what-if forecast created using the CreateWhatIfForecast operation.
Describes a predictor created using the CreateAutoPredictor operation.
Describes an Amazon Forecast dataset created using the CreateDataset operation.
Describes a dataset group created using the CreateDatasetGroup operation.
Describes an dataset import job created using the CreateDatasetImportJob operation.
Describes an Explainability resource created using the CreateExplainability operation.
Describes an Explainability export created using the CreateExplainabilityExport operation.
Describes a forecast created using the CreateForecast operation.
Describes a forecast export job created using the CreateForecastExportJob operation.
Describes a monitor resource.
This operation is only valid for legacy predictors created with CreatePredictor.
Describes a predictor backtest export job created using the CreatePredictorBacktestExport operation.
Describes the what-if analysis created using the CreateWhatIfAnalysis operation.
Describes the what-if forecast created using the CreateWhatIfForecast operation.
Describes the what-if forecast export created using the CreateWhatIfForecastExport operation.
Provides metrics on the accuracy of the models that were trained by the CreatePredictor operation.
Returns a list of dataset groups created using the CreateDatasetGroup operation.
Returns a list of dataset import jobs created using the CreateDatasetImportJob operation.
Returns a list of datasets created using the CreateDataset operation.
Returns a list of Explainability resources created using the CreateExplainability operation.
Returns a list of Explainability exports created using the CreateExplainabilityExport operation.
Returns a list of forecast export jobs created using the CreateForecastExportJob operation.
Returns a list of forecasts created using the CreateForecast operation.
Returns a list of the monitoring evaluation results and predictor events collected by the CreateMonitor operation.
Returns a list of monitors created with the CreateMonitor operation and CreateAutoPredictor operation.
Returns a list of predictor backtest export jobs created using the CreatePredictorBacktestExport operation.
Returns a list of predictors created using the CreateAutoPredictor or CreatePredictor operation.
Lists the tags for an Amazon Forecast resource.
Returns a list of what-if analyses created using the CreateWhatIfAnalysis operation.
Returns a list of what-if forecasts created using the CreateWhatIfForecast operation.
Returns a list of what-if forecasts created using the CreateWhatIfForecastExport operation.
Resumes a stopped monitor resource.
Stops a resource.
Associates the specified tags to a resource with the specified resourceArn.
Deletes the specified tags from a resource.
update_dataset_group

Replaces the datasets in a dataset group with the specified datasets

Examples

```r
## Not run:
svc <- forecastservice()
svc$create_auto_predictor(
  Foo = 123
)

## End(Not run)
```

Amazon Fraud Detector

Description

This is the Amazon Fraud Detector API Reference. This guide is for developers who need detailed information about Amazon Fraud Detector API actions, data types, and errors. For more information about Amazon Fraud Detector features, see the Amazon Fraud Detector User Guide.

We provide the Query API as well as AWS software development kits (SDK) for Amazon Fraud Detector in Java and Python programming languages.

The Amazon Fraud Detector Query API provides HTTPS requests that use the HTTP verb GET or POST and a Query parameter `Action`. AWS SDK provides libraries, sample code, tutorials, and other resources for software developers who prefer to build applications using language-specific APIs instead of submitting a request over HTTP or HTTPS. These libraries provide basic functions that automatically take care of tasks such as cryptographically signing your requests, retrying requests, and handling error responses, so that it is easier for you to get started. For more information about the AWS SDKs, go to Tools to build on AWS page, scroll down to the SDK section, and choose plus (+) sign to expand the section.

Usage

```r
frauddetector(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```
Arguments

```
config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy `https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html`
```

```
credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.
```

```
endpoint Optional shorthand for complete URL to use for the constructed client.
```

```
region Optional shorthand for AWS Region used in instantiating the client.
```

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```
svc <- frauddetector(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
```
frauddetector

secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
),
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

batch_create_variable
batch_get_variable
cancel_batch_import_job
cancel_batch_prediction_job
create_batch_import_job
create_batch_prediction_job
create_detector_version
create_list
create_model
create_model_version
create_rule
create_variable
delete_batch_import_job
delete_batch_prediction_job
delete_detector
delete_detector_version
delete_entity_type
delete_event
delete_events_by_event_type
delete_event_type

Creates a batch of variables
Gets a batch of variables
Cancels an in-progress batch import job
Cancels the specified batch prediction job
Creates a batch import job
Creates a batch prediction job
Creates a detector version
Creates a list
Creates a model using the specified model type
Creates a version of the model using the specified model type and model id
Creates a rule for use with the specified detector
Creates a variable
Deletes the specified batch import job ID record
Deletes a batch prediction job
Deletes the detector
Deletes the detector version
Deletes an entity type
Deletes the specified event
Deletes all events of a particular event type
Deletes an event type
frauddetector

delete_external_model
delete_label
delete_list
delete_model
delete_model_version
delete_outcome
delete_rule
delete_variable
describe_detector
describe_model_versions
get_batch_import_jobs
get_batch_prediction_jobs
get_delete_events_by_event_type_status
get_detectors
get_detector_version
get_entity_types
get_event
get_event_prediction
get_event_prediction_metadata
get_event_types
get_external_models
get_kms_encryption_key
get_labels
get_list_elements
get_lists_metadata
get_models
get_model_version
get_outcomes
get_rules
get_variables
list_event_predictions
list_tags_for_resource
put_detector
put_entity_type
put_event_type
put_external_model
put_kms_encryption_key
put_label
put_outcome
send_event
tag_resource
untag_resource
update_detector_version
update_detector_version_metadata
update_detector_version_status
update_event_label
update_list
update_model

Removes a SageMaker model from Amazon Fraud Detector
Deletes a label
Deletes the list, provided it is not used in a rule
Deletes a model
Deletes a model version
Deletes an outcome
Deletes the rule
Deletes a variable

Gets all versions for a specified detector
Gets all of the model versions for the specified model type or for the specified model
Gets all batch import jobs or a specific job of the specified ID
Gets all batch prediction jobs or a specific job if you specify a job ID
Retrieves the status of a DeleteEventsByEventType action
Gets all detectors or a single detector if a detectorId is specified
Gets a particular detector version
Gets all entity types or a specific entity type if a name is specified
Retrieves details of events stored with Amazon Fraud Detector
Evaluates an event against a detector version
Gets details of the past fraud predictions for the specified event ID, event type, detector ID, and detector version
Gets all event types or a specific event type if name is provided
Gets the details for one or more Amazon SageMaker models that have been imported
Gets the encryption key if a KMS key has been specified to be used to encrypt content
Gets all labels or a specific label if name is provided
Gets all the elements in the specified list
Gets the metadata of either all the lists under the account or the specified list
Gets one or more models
Gets the details of the specified model version
Gets one or more outcomes
Get all rules for a detector (paginated) if ruleId and ruleVersion are not specified
Gets all of the variables or the specific variable
Gets a list of past predictions
Lists all tags associated with the resource
Creates or updates a detector
Creates or updates an entity type
Creates or updates an event type
Creates or updates an Amazon SageMaker model endpoint
Specifies the KMS key to be used to encrypt content in Amazon Fraud Detector
Creates or updates label
Creates or updates an outcome
Stores events in Amazon Fraud Detector without generating fraud predictions for those events
Assigns tags to a resource
Removes tags from a resource
Updates a detector version
Updates the detector version’s description
Updates the detector version’s status
Updates the specified event with a new label
Updates a list
Updates model description
**Description**

Amazon FSx is a fully managed service that makes it easy for storage and application administrators to launch and use shared file storage.

**Usage**

```r
fsx(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

**Arguments**

- `config`: Optional configuration of credentials, endpoint, and/or region.
  
  - `credentials`:
    
      - `creds`:
        
          - `access_key_id`: AWS access key ID
          - `secret_access_key`: AWS secret access key
          - `session_token`: AWS temporary session token
        
        - `profile`: The name of a profile to use. If not given, then the default profile is used.
        
        - `anonymous`: Set anonymous credentials.
      
      - `endpoint`: The complete URL to use for the constructed client.
      
      - `region`: The AWS Region used in instantiating the client.
      
      - `close_connection`: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

• **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoint.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoint.html)

**credentials**
Optional credentials shorthand for the config parameter

• **creds**:
  – **access_key_id**: AWS access key ID
  – **secret_access_key**: AWS secret access key
  – **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**
A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- fsx(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string",
  close_connection = "logical",
  timeout = "numeric",
  s3_force_path_style = "logical",
  sts_regional_endpoint = "string"
)
```
secret_access_key = "string",
session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

associate_file_system_aliases
cancel_data_repository_task
copy_backup
copy_snapshot_and_update_volume
create_backup
create_data_repository_association
create_data_repository_task
create_file_cache
create_file_system
create_file_system_from_backup
create_snapshot
create_storage_virtual_machine
create_volume
create_volume_from_backup
delete_backup
delete_data_repository_association
delete_file_cache
delete_file_system
delete_snapshot
delete_storage_virtual_machine
delete_volume
describe_backups
describe_data_repository_associations
describe_data_repository_tasks
describe_file_caches
describe_file_system_aliases
describe_file_systems
describe_shared_vpc_configuration
describe_snapshots
describe_storage_virtual_machines
describe_volumes
disassociate_file_system_aliases
list_tags_for_resource
release_file_system_nfs_v3_locks
restore_volume_from_snapshot
start_misconfigured_state_recovery

Use this action to associate one or more Domain Name Server (DNS) aliases with an Amazon FSx for Windows File Server file system.

Cancels an existing Amazon FSx for Lustre data repository task if that task is in either the PENDING or EXECUTING state.

Copies an existing backup within the same Amazon Web Services account to another Amazon Web Services Region.

Updates an existing volume by using a snapshot from another Amazon FSx for OpenZFS file system.

Creates a backup of an existing Amazon FSx for Windows File Server file system, Amazon FSx for Lustre file system, Amazon FSx for NetApp ONTAP volume, or Amazon FSx for OpenZFS file system.

Creates an Amazon FSx for Lustre data repository association (DRA).

Creates an Amazon FSx for Lustre data repository task.

Creates a new Amazon File Cache resource.

Creates a new, empty Amazon FSx file system.

Creates a new Amazon FSx for Lustre, Amazon FSx for Windows File Server, or Amazon FSx for OpenZFS file system.

Creates a snapshot of an existing Amazon FSx for OpenZFS volume.

Creates a storage virtual machine (SVM) for an Amazon FSx for ONTAP file system.

A new Amazon FSx for NetApp ONTAP volume from an existing Amazon FSx for OpenZFS volume.

Deletes an Amazon FSx backup.

Deletes a data repository association on an Amazon FSx for Lustre file system.

Deletes an Amazon File Cache resource.

Deletes a file system.

Deletes an Amazon FSx for OpenZFS snapshot.

Deletes an existing Amazon FSx for ONTAP storage virtual machine (SVM).

Deletes an Amazon FSx for NetApp ONTAP or Amazon FSx for OpenZFS volume.

Returns the description of a specific Amazon FSx backup, if a BackupIds value is present.

Returns the description of specific Amazon FSx for Lustre or Amazon File Cache data repository tasks, if one or more TaskIds values are present.

Returns the description of specific Amazon FSx for Lustre or Amazon File Cache data repository associations, if one or more AssociationIds values are present.

Returns the description of specific Amazon FSx file systems, if a FileSystemIds value is present.

Indicates whether participant accounts in your organization can create Amazon FSx for NetApp ONTAP Multi-AZ file systems in subnets that are shared by a virtual private cloud (VPC) owner.

Returns the description of specific Amazon FSx for OpenZFS snapshots, if a SnapshotIds value is present.

Describes one or more Amazon FSx for NetApp ONTAP storage virtual machines (SVMs).

Describes one or more Amazon FSx for NetApp ONTAP or Amazon FSx for OpenZFS file systems.

Use this action to disassociate, or remove, one or more Domain Name Service (DNS) aliases from an Amazon FSx for Windows File Server file system.

Lists tags for Amazon FSx resources.

Releases the file system lock from an Amazon FSx for OpenZFS file system.

Returns an Amazon FSx for OpenZFS volume to the state saved by the specified snapshot.

After performing steps to repair the Active Directory configuration of an FSx for Windows File Server file system.
The `tag_resource` command tags an Amazon FSx resource. The `untag_resource` command removes a tag from an Amazon FSx resource. The `update_data_repository_association` command updates the configuration of an existing data repository association on an Amazon FSx file system. The `update_file_cache` command updates the configuration of an existing Amazon File Cache resource. The `update_file_system` command updates the configuration of an existing Amazon FSx file system. The `update_shared_vpc_configuration` command configures whether participant accounts in your organization can create Amazon FSx for NetApp ONTAP Multi-AZ file systems in subnets that are shared by a virtual private cloud (VPC). The `update_snapshot` command updates the name of an Amazon FSx for OpenZFS snapshot. The `update_storage_virtual_machine` command updates an FSx for ONTAP storage virtual machine (SVM). The `update_volume` command updates the configuration of an Amazon FSx for NetApp ONTAP or Amazon FSx for OpenZFS volume.

### Examples

```r
## Not run:
svc <- fsx()
# This operation copies an Amazon FSx backup.
svc$copy_backup(
  SourceBackupId = "backup-03e3c82e0183b7b6b",
  SourceRegion = "us-east-2"
)
## End(Not run)
```

---

**glacier**

`Amazon Glacier`

## Description

Amazon S3 Glacier (Glacier) is a storage solution for "cold data."

Glacier is an extremely low-cost storage service that provides secure, durable, and easy-to-use storage for data backup and archival. With Glacier, customers can store their data cost effectively for months, years, or decades. Glacier also enables customers to offload the administrative burdens of operating and scaling storage to AWS, so they don’t have to worry about capacity planning, hardware provisioning, data replication, hardware failure and recovery, or time-consuming hardware migrations.

Glacier is a great storage choice when low storage cost is paramount and your data is rarely retrieved. If your application requires fast or frequent access to your data, consider using Amazon S3. For more information, see [Amazon Simple Storage Service (Amazon S3)](https://aws.amazon.com/s3/).

You can store any kind of data in any format. There is no maximum limit on the total amount of data you can store in Glacier.

If you are a first-time user of Glacier, we recommend that you begin by reading the following sections in the [Amazon S3 Glacier Developer Guide](https://docs.aws.amazon.com/AmazonS3/latest/userguide/what-is-glacier.html):
• What is Amazon S3 Glacier - This section of the Developer Guide describes the underlying data model, the operations it supports, and the AWS SDKs that you can use to interact with the service.

• Getting Started with Amazon S3 Glacier - The Getting Started section walks you through the process of creating a vault, uploading archives, creating jobs to download archives, retrieving the job output, and deleting archives.

Usage

```
glacier(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

config Optional configuration of credentials, endpoint, and/or region.

- credentials:
  - creds:
    * access_key_id: AWS access key ID
    * secret_access_key: AWS secret access key
    * session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.

- endpoint: The complete URL to use for the constructed client.

- region: The AWS Region used in instantiating the client.

- close_connection: Immediately close all HTTP connections.

- timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

- s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

- sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoint.html

credentials Optional credentials shorthand for the config parameter

- creds:
  - access_key_id: AWS access key ID
  - secret_access_key: AWS secret access key
  - session_token: AWS temporary session token

- profile: The name of a profile to use. If not given, then the default profile is used.

- anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.
Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- glacier(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- **abort_multipart_upload**: This operation aborts a multipart upload identified by the upload ID
- **abort_vault_lock**: This operation aborts the vault locking process if the vault lock is not in the Locked state
- **add_tags_to_vault**: This operation adds the specified tags to a vault
- **complete_multipart_upload**: You call this operation to inform Amazon S3 Glacier (Glacier) that all the archive parts have been uploaded
- **complete_vault_lock**: This operation completes the vault locking process by transitioning the vault lock from the InProgress state to the Locked state
- **create_vault**: This operation creates a new vault with the specified name
- **delete_archive**: This operation deletes an archive from a vault
- **delete_vault**: This operation deletes a vault
delete_vault_access_policy  This operation deletes the access policy associated with the specified vault
delete_vault_notifications  This operation deletes the notification configuration set for a vault
describe_job  This operation returns information about a job you previously initiated, including the job initiation date, status, and output
describe_vault  This operation returns information about a vault, including the vault’s Amazon Resource Name (ARN), the date the vault was created, the number of archives it contains, and the total size of all the archives in the vault
get_data_retrieval_policy  This operation downloads the output of the job you initiated using InitiateJob
get_job_output  This operation retrieves the access-policy subresource set on the vault; for more information on setting this subresource, see Set Vault Access Policy (PUT access-policy)
get_vault_access_policy  This operation retrieves the following attributes from the lock-policy subresource set on the vault:
get_vault_lock  This operation retrieves the notification-configuration subresource of the specified vault
get_vault_notifications  This operation retrieves a job of the specified type, which can be a select, an archival retrieval, or a vault retrieval
initiate_job  This operation initiates a multipart upload
initiate_multipart_upload  This operation initiates the vault locking process by doing the following:
initiate_vault_lock  This operation lists jobs for a vault, including jobs that are in-progress and jobs that have recently finished
list_jobs  This operation lists in-progress multipart uploads for the specified vault
list_multipart_uploads  This operation lists the parts of an archive that have been uploaded in a specific multipart upload
list_parts  This operation lists the provisioned capacity units for the specified AWS account
list_provisioned_capacity  This operation lists all the tags attached to a vault
list_tags_for_vault  This operation lists all vaults owned by the calling user’s account
list_vaults  This operation purchases a provisioned capacity unit for an AWS account
purchase_provisioned_capacity  This operation lists in-progress multipart uploads for the specified vault
remove_tags_from_vault  This operation removes one or more tags from the set of tags attached to a vault
set_data_retrieval_policy  This operation lists in-progress multipart uploads for the specified vault
set_vault_access_policy  This operation configures an access policy for a vault and will overwrite an existing policy
set_vault_notifications  This operation configures notifications that will be sent when specific events happen to a vault
upload_archive  This operation adds an archive to a vault
upload_multipart_part  This operation uploads a part of an archive
 upload_multipart_part

Examples

```r
## Not run:
svc <- glacier()
# The example deletes an in-progress multipart upload to a vault named
# my-vault:
svc$abort_multipart_upload(
  accountId = "-",
  uploadId = "19gaReZEXAMPLES6Ry5YYqthHOC_kGRCT03L9yetr220UmPtBYKk-OssZtLq...",
  vaultName = "my-vault"
)
## End(Not run)
```

---

globalaccelerator  AWS Global Accelerator
Description

Global Accelerator

This is the *Global Accelerator API Reference*. This guide is for developers who need detailed information about Global Accelerator API actions, data types, and errors. For more information about Global Accelerator features, see the *Global Accelerator Developer Guide*.

Global Accelerator is a service in which you create *accelerators* to improve the performance of your applications for local and global users. Depending on the type of accelerator you choose, you can gain additional benefits.

- By using a standard accelerator, you can improve availability of your internet applications that are used by a global audience. With a standard accelerator, Global Accelerator directs traffic to optimal endpoints over the Amazon Web Services global network.
- For other scenarios, you might choose a custom routing accelerator. With a custom routing accelerator, you can use application logic to directly map one or more users to a specific endpoint among many endpoints.

Global Accelerator is a global service that supports endpoints in multiple Amazon Web Services Regions but you must specify the US West (Oregon) Region to create, update, or otherwise work with accelerators. That is, for example, specify `--region us-west-2` on Amazon Web Services CLI commands.

By default, Global Accelerator provides you with static IP addresses that you associate with your accelerator. The static IP addresses are anycast from the Amazon Web Services edge network. For IPv4, Global Accelerator provides two static IPv4 addresses. For dual-stack, Global Accelerator provides a total of four addresses: two static IPv4 addresses and two static IPv6 addresses. With a standard accelerator for IPv4, instead of using the addresses that Global Accelerator provides, you can configure these entry points to be IPv4 addresses from your own IP address ranges that you bring to Global Accelerator (BYOIP).

For a standard accelerator, they distribute incoming application traffic across multiple endpoint resources in multiple Amazon Web Services Regions, which increases the availability of your applications. Endpoints for standard accelerators can be Network Load Balancers, Application Load Balancers, Amazon EC2 instances, or Elastic IP addresses that are located in one Amazon Web Services Region or multiple Amazon Web Services Regions. For custom routing accelerators, you map traffic that arrives to the static IP addresses to specific Amazon EC2 servers in endpoints that are virtual private cloud (VPC) subnets.

The static IP addresses remain assigned to your accelerator for as long as it exists, even if you disable the accelerator and it no longer accepts or routes traffic. However, when you *delete* an accelerator, you lose the static IP addresses that are assigned to it, so you can no longer route traffic by using them. You can use IAM policies like tag-based permissions with Global Accelerator to limit the users who have permissions to delete an accelerator. For more information, see *Tag-based policies*.

For standard accelerators, Global Accelerator uses the Amazon Web Services global network to route traffic to the optimal regional endpoint based on health, client location, and policies that you configure. The service reacts instantly to changes in health or configuration to ensure that internet traffic from clients is always directed to healthy endpoints.

For more information about understanding and using Global Accelerator, see the *Global Accelerator Developer Guide*. 
globalaccelerator

Usage

globalaccelerator(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      • access_key_id: AWS access key ID
      • secret_access_key: AWS secret access key
      • session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

dendpoint Optional shorthand for complete URL to use for the constructed client.

dregion Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```r
svc <- globalaccelerator(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- **add_custom_routing_endpoints**: Associate a virtual private cloud (VPC) subnet endpoint with your custom routing accelerator.
- **add_endpoints**: Add endpoints to an endpoint group.
- **advertise_byoip_cidr**: Advertises an IPv4 address range that is provisioned for use with your Amazon Web Services resources through bring your own IP addresses (BYOIP).
- **allow_custom_routing_traffic**: Specify the Amazon EC2 instance (destination) IP addresses and ports that can receive traffic for a custom routing accelerator.
- **create_accelerator**: Create a custom routing accelerator.
- **create_cross_account_attachment**: Create a cross-account attachment in Global Accelerator.
- **create_custom_routing_accelerator**: Create a custom routing accelerator.
- **create_custom_routing_endpoint_group**: Create an endpoint group for the specified listener for a custom routing accelerator.
- **create_custom_routing_listener**: Create a listener to process inbound connections from clients to a custom routing accelerator.
- **create_endpoint_group**: Create an endpoint group for the specified listener.
- **create_listener**: Create a listener to process inbound connections from clients to an accelerator.
- **delete_accelerator**: Delete an accelerator.
- **delete_cross_account_attachment**: Delete a cross-account attachment.
- **delete_custom_routing_accelerator**: Delete a custom routing accelerator.
delete_custom_routing_endpoint_group
delete_custom_routing_listener
delete_endpoint_group
delete_listener
deny_custom_routing_traffic
deprovision_byoip_cidr
describe_accelerator
describe_accelerator_attributes
describe_cross_account_attachment
describe_custom_routing_accelerator
describe_custom_routing_accelerator_attributes
describe_custom_routing_endpoint_group
describe_custom_routing_endpoint_listener
describe_endpoint_group
describe_listener
list_accelerators
list_byoip_cidrs
list_cross_account_attachments
list_cross_account_resource_accounts
list_cross_account_resources
list_custom_routing_accelerators
list_custom_routing_endpoint_groups
list_custom_routing_listeners
list_custom_routing_portMappings
list_custom_routing_portMappings_by_destination
list_endpoint_groups
list_listeners
list_tags_for_resource
provision_byoip_cidr
remove_custom_routing_endpoints
remove_endpoints
tag_resource
untag_resource
update_accelerator
update_accelerator_attributes
update_cross_account_attachment
update_custom_routing_accelerator
update_custom_routing_accelerator_attributes
update_custom_routing_listener
update_endpoint_group
update_listener
withdraw_byoip_cidr

Delete an endpoint group from a listener for a custom routing accelerator.
Delete a listener for a custom routing accelerator.
Delete an endpoint group from a listener.
Delete a listener from an accelerator.
Specify the Amazon EC2 instance (destination) IP addresses and ports
Releases the specified address range that you provisioned to use with your accelerator.
Describe an accelerator.
Describe the attributes of an accelerator.
Gets configuration information about a cross-account attachment.
Describe a custom routing accelerator.
Describe the attributes of a custom routing accelerator.
Describe an endpoint group for a custom routing accelerator.
The description of a listener for a custom routing accelerator.
Describe an endpoint group.
Describe a listener.
List the accelerators for an Amazon Web Services account.
Lists the IP address ranges that were specified in calls to ProvisionByoipCidr.
List the cross-account attachments that have been created in Global Accelerator.
List the accounts that have cross-account resources.
List the cross-account resources available to work with.
List the custom routing accelerators for an Amazon Web Services account.
List the endpoint groups that are associated with a listener for a custom routing accelerator.
List the listeners for a custom routing accelerator.
Provides a complete mapping from the public accelerator IP address and port to the destination EC2 instance IP addresses and ports.
List the port mappings for a specific EC2 instance (destination) in a VPC subnet endpoint.
List the endpoint groups that are associated with a listener for an accelerator.
List all tags for an accelerator.
Provisions an IP address range to use with your Amazon Web Services resources.
Remove endpoints from a custom routing accelerator.
Remove endpoints from an endpoint group.
Add tags to an accelerator resource.
Remove tags from a Global Accelerator resource.
Update an accelerator to make changes, such as the following:
Update the attributes for an accelerator.
Update a cross-account attachment to add or remove principals or resources.
Update a custom routing accelerator.
Update the attributes for a custom routing accelerator.
Update a listener for a custom routing accelerator.
Update an endpoint group.
Update a listener.
Stops advertising an address range that is provisioned as an address pool.

Examples

```r
## Not run:
svc <- globalaccelerator()
```


```python
svc$add_custom_routing_endpoints(
    Foo = 123
)

## End(Not run)
```

---

**glue**

**AWS Glue**

### Description

Glue

Defines the public endpoint for the Glue service.

### Usage

```r
glue(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

### Arguments

- **config**
  - Optional configuration of credentials, endpoint, and/or region.
    - **credentials**
      - **creds**:
        - **access_key_id**: AWS access key ID
        - **secret_access_key**: AWS secret access key
        - **session_token**: AWS temporary session token
        - **profile**: The name of a profile to use. If not given, then the default profile is used.
        - **anonymous**: Set anonymous credentials.
    - **endpoint**: The complete URL to use for the constructed client.
    - **region**: The AWS Region used in instantiating the client.
    - **close_connection**: Immediately close all HTTP connections.
    - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
    - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
    - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

- **credentials**
  - Optional credentials shorthand for the config parameter
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
– **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**
A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- glue(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string",
  close_connection = "logical",
  timeout = "numeric",
  s3_force_path_style = "logical",
  sts_regional_endpoint = "string"
),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

**Operations**
batch_create_partition
batch_delete_connection
batch_delete_partition
batch_delete_table
batch_delete_table_version
batch_get_blueprints
batch_get_crawlers
batch_get_custom_entity_types
batch_get_data_quality_result
batch_get_dev_endpoints
batch_get_jobs
batch_get_partition
batch_get_table_optimizer
batch_get_triggers
batch_get_workflows
batch_stop_job_run
batch_update_partition
cancel_data_quality_rule_recommendation_run
cancel_data_quality_ruleset_evaluation_run
cancel_ml_task_run
cancel_statement
check_schema_version_validity
create_blueprint
create_classifier
create_connection
create_crawler
create_custom_entity_type
create_database
create_data_quality_ruleset
create_dev_endpoint
create_job
create_ml_transform
create_partition
create_partition_index
create_registry
create_schema
create_script
create_security_configuration
create_session
create_table
create_table_optimizer
create_trigger
create_user_defined_function
create_workflow
delete_blueprint
delete_classifier
delete_column_statistics_for_partition
delete_column_statistics_for_table

Creating one or more partitions in a batch operation
Deletes a list of connection definitions from the Data Catalog
Deletes one or more partitions in a batch operation
Deletes multiple tables at once
Deletes a specified batch of versions of a table
Retrieves information about a list of blueprints
Retrieves a list of resource metadata for a given list of crawler names
Retrieves the details for the custom patterns specified by a list of names
Retrieves a list of data quality results for the specified result IDs
Retrieves a list of resource metadata for a given list of development endpoint names
Retrieves a list of resource metadata for a given list of job names
Retrieves partitions in a batch request
Retrieves the configuration for the specified table optimizers
Retrieves a list of resource metadata for a given list of trigger names
Retrieves a list of resource metadata for a given list of workflow names
Stops one or more job runs for a specified job definition
Updates one or more partitions in a batch operation
Cancels the specified recommendation run that was being used to generate recommendations
Cancels a run where a ruleset is being evaluated against a data source
Cancels (stops) a task run
Cancels the statement
Validates the supplied schema
Registers a blueprint with Glue
Creates a classifier in the user’s account
Creates a connection definition in the Data Catalog
Creates a new crawler with specified targets, role, configuration, and optional schedule
Creates a custom pattern that is used to detect sensitive data across the columns in a table
Creates a new database in the Data Catalog
Creates a data quality ruleset with DQDL rules applied to a specified Glue table
Creates a new development endpoint
Creates a new job definition
Creates an Glue machine learning transform
Creates a new partition
Creates a specified partition index in an existing table
Creates a new registry which may be used to hold a collection of schemas
Creates a new schema set and registers the schema definition
Transforms a directed acyclic graph (DAG) into code
Creates a new security configuration
Creates a new session
Creates a new table definition in the Data Catalog
Creates a new table optimizer for a specific function
Creates a new trigger
Creates a new function definition in the Data Catalog
Creates a new workflow
Deletes an existing blueprint
Removes a classifier from the Data Catalog
Delete the partition column statistics of a column
Retrieves table statistics of columns
delete_connection
delete_crawler
delete_custom_entity_type
delete_database
delete_data_quality_ruleset
delete_dev_endpoint
delete_job
delete_ml_transform
delete_partition
delete_partition_index
delete_registry
delete_resource_policy
delete_schema
delete_schema_versions
delete_security_configuration
delete_session
delete_table
delete_table_optimizer
delete_table_version
delete_trigger
delete_user_defined_function
delete_workflow
get_blueprint
get_blueprint_run
get_blueprint_runs
get_catalog_import_status
get_classifier
get_classifiers
get_column_statistics_for_partition
get_column_statistics_for_table
get_column_statistics_task_run
get_column_statistics_task_runs
get_connection
get_connections
get_crawler
get_crawler_metrics
crawlers
get_custom_entity_type
get_database
get_databases
get_data_catalog_encryption_settings
get_dataflow_graph
get_data_quality_result
get_data_quality_rule_recommendation_run
get_data_quality_ruleset
get_data_quality_ruleset_evaluation_run
get_dev_endpoint
get_dev_endpoints

Deletes a connection from the Data Catalog
Removes a specified crawler from the Glue Data Catalog, unless the crawler is RUNNING
Deletes a custom pattern by specifying its name
Removes a specified database from a Data Catalog
Deletes a data quality ruleset
Deletes a specified development endpoint
Deletes a specified job definition
Deletes an Glue machine learning transform
Deletes a specified partition
Deletes a specified partition index from an existing table
Delete the entire registry including schema and all of its versions
Deletes a specified policy
Deletes the entire schema set, including the schema set and all of its versions
Remove versions from the specified schema
Deletes a specified security configuration
Deletes the session
Removes a table definition from the Data Catalog
Deletes an optimizer and all associated metadata for a table
Deletes a specified version of a table
Deletes a specified trigger
Deletes an existing function definition from the Data Catalog
Deletes a workflow
Retrieves the details of a blueprint
Retrieves the details of a blueprint run
Retrieves the details of blueprint runs for a specified blueprint
Retrieves the status of a migration operation
Retrieve a classifier by name
Lists all classifier objects in the Data Catalog
Retrieves partition statistics of columns
Retrieves table statistics of columns
Get the associated metadata/information for a task run, given a task run ID
Retrieves information about all runs associated with the specified table
Retrieves a connection definition from the Data Catalog
Retrieves a list of connection definitions from the Data Catalog
Retrieves metadata for a specified crawler
Retrieves metrics about specified crawlers
Retrieves metadata for all crawlers defined in the customer account
Retrieves the details of a custom pattern by specifying its name
Retrieves the definition of a specified database
Retrieves all databases defined in a given Data Catalog
Retrieves the security configuration for a specified catalog
Transforms a Python script into a directed acyclic graph (DAG)
Retrieves the result of a data quality rule evaluation
Gets the specified recommendation run that was used to generate rules
Returns an existing ruleset by identifier or name
Retrieves a specific run where a ruleset is evaluated against a data source
Retrieves information about a specified development endpoint
Retrieves all the development endpoints in this Amazon Web Services account
get_job
get_job_bookmark
get_job_run
get_job_runs
get_jobs
get_mapping
get_ml_task_run
get_ml_task_runs
get_ml_transform
get_ml_transforms
get_partition
get_partition_indexes
get_partitions
get_plan
get_registry
get_resource_policies
get_resource_policy
get_schema
get_schema_by_definition
get_schema_version
get_schema_versions_diff
get_security_configuration
get_security_configurations
get_session
get_statement
get_table
get_table_optimizer
get_tables
get_table_version
get_table_versions
get_tags
get_trigger
get_triggers
get_unfiltered_partition_metadata
get_unfiltered_partitions_metadata
get_unfiltered_table_metadata
get_user_defined_function
get_user_defined_functions
get_workflow
get_workflow_run
get_workflow_run_properties
get_workflow_runs
import_catalog_to_glue
list_blueprints
list_column_statistics_task_runs
list_crawlers
list_crawls
list_custom_entity_types

get_job
get_job_bookmark
get_job_run
get_job_runs
get_jobs
get_mapping
get_ml_task_run
get_ml_task_runs
get_ml_transform
get_ml_transforms
get_partition
get_partition_indexes
get_partitions
get_plan
get_registry
get_resource_policies
get_resource_policy
get_schema
get_schema_by_definition
get_schema_version
get_schema_versions_diff
get_security_configuration
get_security_configurations
get_session
get_statement
get_table
get_table_optimizer
get_tables
get_table_version
get_table_versions
get_tags
get_trigger
get_triggers
get_unfiltered_partition_metadata
get_unfiltered_partitions_metadata
get_unfiltered_table_metadata
get_user_defined_function
get_user_defined_functions
get_workflow
get_workflow_run
get_workflow_run_properties
get_workflow_runs
import_catalog_to_glue
list_blueprints
list_column_statistics_task_runs
list_crawlers
list_crawls
list_custom_entity_types

Retrieves an existing job definition
Retrieves information on a job bookmark entry
Retrieves the metadata for a given job run
Retrieves metadata for all runs of a given job definition
Retrieves all current job definitions
Creates mappings
Gets details for a specific task run on a machine learning transform
Gets a list of runs for a machine learning transform
Gets an Glue machine learning transform artifact and all its corresponding run information
Gets a sortable, filterable list of existing Glue machine learning transforms
Retrieves information about a specified partition
Retrieves the partition indexes associated with a table
Retrieves information about the partitions in a table
Gets code to perform a specified mapping
Describes the specified registry in detail
Retrieves the resource policies set on individual resources by Resource Access Manager
Retrieves a specified resource policy
Describes the specified schema in detail
Retrieves a schema by the SchemaDefinition
Get the specified schema by its unique ID assigned when a version of the schema was created
Fetches the schema version difference in the specified difference type between two versions of a schema
Retrieves a specified security configuration
Retrieves a list of all security configurations
Retrieves the session
Retrieves the statement
Retrieves the Table definition in a Data Catalog for a specified table
Returns the configuration of all optimizers associated with a specified table
Retrieves the definitions of some or all of the tables in a given Database
Retrieves a specified version of a table
Retrieves a list of strings that identify available versions of a specified table
Retrieves a list of tags associated with a resource
Retrieves the definition of a trigger
Gets all the triggers associated with a job
Retrieves partition metadata from the Data Catalog that contains unfiltered metadata
Retrieves partition metadata from the Data Catalog that contains unfiltered metadata
Allows a third-party analytical engine to retrieve unfiltered table metadata from a Data Catalog
Retrieves a specified function definition from the Data Catalog
Retrieves multiple function definitions from the Data Catalog
Retrieves resource metadata for a workflow
Retrieves the metadata for a given workflow run
Retrieves the workflow run properties which were set during the run
Retrieves metadata for all runs of a given workflow
Imports an existing Amazon Athena Data Catalog to Glue
Lists all the blueprint names in an account
List all task runs for a particular account
Retrieves the names of all crawler resources in this Amazon Web Services account
Returns all the crawls of a specified crawler
Lists all the custom patterns that have been created
list_data_quality_results
list_data_quality_rule_recommendation_runs
list_data_quality_ruleset_evaluation_runs
list_data_quality_rule_sets
list_dev_endpoints
list_jobs
list_ml_transforms
list_registries
list_schemas
list_schema_versions
list_stages
list_table_optimizer_runs
list_triggers
list_workflows
put_data_catalog_encryption_settings
put_resource_policy
put_schema_version_metadata
put_workflow_run_properties
query_schema_version_metadata
register_schema_version
remove_schema_version_metadata
reset_job_bookmark
resume_workflow_run
run_statement
search_tables
start_blueprint_run
start_column_statistics_task_run
start_crawler
start_crawler_schedule
start_data_quality_rule_recommendation_run
start_data_quality_ruleset_evaluation_run
start_export_labels_task_run
start_import_labels_task_run
start_job_run
start_ml_evaluation_task_run
start_ml_labeling_set_generation_task_run
start_trigger
start_workflow_run
stop_column_statistics_task_run
stop_crawler
stop_crawler_schedule
stop_session
stop_trigger
stop_workflow_run
tag_resource
untag_resource
update_blueprint

Returns all data quality execution results for your account
Lists the recommendation runs meeting the filter criteria
Lists all the runs meeting the filter criteria, where a ruleset is evaluated against a data source
Returns a paginated list of rulesets for the specified list of Glue tables
Retrieves the names of all DevEndpoint resources in this Amazon Web Services account
Retrieves the names of all job resources in this Amazon Web Services account
Retrieves a sortable, filterable list of existing Glue machine learning transforms in this Amazon Web Services account
Retains a list of registries that you have created, with minimal registry information
Retains a list of schemas with minimal details
Retains a list of schema versions that you have created, with minimal information
Retrieve a list of sessions
Lists statements for the session
Lists the history of previous optimizer runs for a specific table
Retrieves the names of all trigger resources in this Amazon Web Services account
Lists names of workflows created in the account
Sets the security configuration for a specified catalog
Sets the Data Catalog resource policy for access control
Puts the metadata key value pair for a specified schema version ID
Puts the specified workflow run properties for the given workflow run
Queries for the schema version metadata information
Adds a new version to the existing schema
Removes a key value pair from the schema version metadata for the specified schema version ID
Resets a bookmark entry
Restarts selected nodes of a previous partially completed workflow run and resumes the workflow run
Executes the statement
 Searches a set of tables based on properties in the table metadata as well as on the parent database
Starts a new run of the specified blueprint
Starts a column statistics task run, for a specified table and columns
Starts a crawl using the specified crawler, regardless of what is scheduled
Changes the schedule state of the specified crawler to SCHEDULED, unless the crawler is already running
Starts a recommendation run that is used to generate rules when you don’t know what rules to write
Once you have a ruleset definition (either recommended or your own), you can call this operation
Begins an asynchronous task to export all labeled data for a particular transform
Enables you to provide additional labels (examples of truth) to be used to train
Starts a job run using a job definition
Starts a task to estimate the quality of the transform
Starts the active learning workflow for your machine learning transform to improve
Starts an existing trigger
Starts a new run of the specified workflow
Stops a task run for the specified table
If the specified crawler is running, stops the crawl
Sets the schedule state of the specified crawler to NOT_SCHEDULED, but does not stop the crawler if it is already running
Stops the session
Stops a specified trigger
Stops the execution of the specified workflow run
Adds tags to a resource
Removes tags from a resource
Updates a registered blueprint
update_classifier  
update_column_statistics_for_partition  
update_column_statistics_for_table  
update_connection  
update_crawler  
update_crawler_schedule  
update_database  
update_data_quality_ruleset  
update_dev_endpoint  
update_job  
update_job_from_source_control  
update_ml_transform  
update_partition  
update_registry  
update_schema  
update_source_control_from_job  
update_table  
update_table_optimizer  
update_trigger  
update_user_defined_function  
update_workflow

Modifies an existing classifier (a GrokClassifier, an XMLClassifier, a JsonClassifier, or a CsvClassifier, depending on which field is present)

Creates or updates partition statistics of columns

Updates a connection definition in the Data Catalog

Updates a crawler

Updates the schedule of a crawler using a cron expression

Updates an existing database definition in a Data Catalog

Updates the specified data quality ruleset

Updates a specified development endpoint

Updates an existing job definition

Synchronizes a job from the source control repository

Updates an existing machine learning transform

Updates a partition

Updates an existing registry which is used to hold a collection of schemas

Updates the description, compatibility setting, or version checkpoint for a schema

Synchronizes a job to the source control repository

Updates a metadata table in the Data Catalog

Updates the configuration for an existing table optimizer

Updates a trigger definition

Updates an existing function definition in the Data Catalog

Updates an existing workflow

---

**Examples**

```r
## Not run:
svc <- glue()
svc$batch_create_partition(  
  Foo = 123  
)

## End(Not run)
```

---

**Description**

Glue DataBrew is a visual, cloud-scale data-preparation service. DataBrew simplifies data preparation tasks, targeting data issues that are hard to spot and time-consuming to fix. DataBrew empowers users of all technical levels to visualize the data and perform one-click data transformations, with no coding required.
Usage

```r
gluedatabrew(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

- **endpoint**: Optional shorthand for complete URL to use for the constructed client.

- **region**: Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```r
svc <- gluedatabrew(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- **batch_delete_recipe_version**: Deletes one or more versions of a recipe at a time
- **create_dataset**: Creates a new DataBrew dataset
- **create_profile_job**: Creates a new job to analyze a dataset and create its data profile
- **create_project**: Creates a new DataBrew project
- **create_recipe**: Creates a new DataBrew recipe
- **create_recipe_job**: Creates a new job to transform input data, using steps defined in an existing Glue DataBrew recipe
- **create_ruleset**: Creates a new ruleset that can be used in a profile job to validate the data quality of a dataset
- **create_schedule**: Creates a new schedule for one or more DataBrew jobs
- **delete_dataset**: Deletes a dataset from DataBrew
- **delete_job**: Deletes the specified DataBrew job
- **delete_project**: Deletes an existing DataBrew project
- **delete_recipe_version**: Deletes a single version of a DataBrew recipe
- **delete_ruleset**: Deletes a ruleset
- **delete_schedule**: Deletes the specified DataBrew schedule
<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>describe_dataset</code></td>
<td>Returns the definition of a specific DataBrew dataset</td>
</tr>
<tr>
<td><code>describe_job</code></td>
<td>Returns the definition of a specific DataBrew job</td>
</tr>
<tr>
<td><code>describe_job_run</code></td>
<td>Represents one run of a DataBrew job</td>
</tr>
<tr>
<td><code>describe_project</code></td>
<td>Returns the definition of a specific DataBrew project</td>
</tr>
<tr>
<td><code>describe_recipe</code></td>
<td>Returns the definition of a specific DataBrew recipe corresponding to a particular version</td>
</tr>
<tr>
<td><code>describe_ruleset</code></td>
<td>Retrieves detailed information about the ruleset</td>
</tr>
<tr>
<td><code>describe_schedule</code></td>
<td>Returns the definition of a specific DataBrew schedule</td>
</tr>
<tr>
<td><code>list_datasets</code></td>
<td>Lists all of the DataBrew datasets</td>
</tr>
<tr>
<td><code>list_job_runs</code></td>
<td>Lists all of the previous runs of a particular DataBrew job</td>
</tr>
<tr>
<td><code>list_jobs</code></td>
<td>Lists all of the DataBrew jobs that are defined</td>
</tr>
<tr>
<td><code>list_projects</code></td>
<td>Lists all of the DataBrew projects that are defined</td>
</tr>
<tr>
<td><code>list_recipes</code></td>
<td>Lists all of the DataBrew recipes that are defined</td>
</tr>
<tr>
<td><code>list_recipe_versions</code></td>
<td>Lists the versions of a particular DataBrew recipe, except for LATEST_WORKING</td>
</tr>
<tr>
<td><code>list_rulesets</code></td>
<td>List all rulesets available in the current account or rulesets associated with a specific resource (optional)</td>
</tr>
<tr>
<td><code>list_schedules</code></td>
<td>Lists the DataBrew schedules that are defined</td>
</tr>
<tr>
<td><code>list_tags_for_resource</code></td>
<td>Lists all the tags for a DataBrew resource</td>
</tr>
<tr>
<td><code>publish_recipe</code></td>
<td>Publishes a new version of a DataBrew recipe</td>
</tr>
<tr>
<td><code>send_project_session_action</code></td>
<td>Performs a recipe step within an interactive DataBrew session that’s currently open</td>
</tr>
<tr>
<td><code>start_job_run</code></td>
<td>Runs a DataBrew job</td>
</tr>
<tr>
<td><code>start_project_session</code></td>
<td>Creates an interactive session, enabling you to manipulate data in a DataBrew project</td>
</tr>
<tr>
<td><code>stop_job_run</code></td>
<td>Stops a particular run of a job</td>
</tr>
<tr>
<td><code>tag_resource</code></td>
<td>Adds metadata tags to a DataBrew resource, such as a dataset, project, recipe, job, or schedule</td>
</tr>
<tr>
<td><code>untag_resource</code></td>
<td>Removes metadata tags from a DataBrew resource</td>
</tr>
<tr>
<td><code>update_dataset</code></td>
<td>Modifies the definition of an existing DataBrew dataset</td>
</tr>
<tr>
<td><code>update_profile_job</code></td>
<td>Modifies the definition of an existing profile job</td>
</tr>
<tr>
<td><code>update_project</code></td>
<td>Modifies the definition of an existing DataBrew project</td>
</tr>
<tr>
<td><code>update_recipe</code></td>
<td>Modifies the definition of the LATEST_WORKING version of a DataBrew recipe</td>
</tr>
<tr>
<td><code>update_recipe_job</code></td>
<td>Modifies the definition of an existing DataBrew recipe job</td>
</tr>
<tr>
<td><code>update_ruleset</code></td>
<td>Updates specified ruleset</td>
</tr>
<tr>
<td><code>update_schedule</code></td>
<td>Modifies the definition of an existing DataBrew schedule</td>
</tr>
</tbody>
</table>

**Examples**

```r
## Not run:
svc <- gluedatabrew()
svc$batch_delete_recipe_version(
  Foo = 123
)
## End(Not run)
```

---

guardduty  Amazon GuardDuty
Description

Amazon GuardDuty is a continuous security monitoring service that analyzes and processes the following foundational data sources - VPC flow logs, Amazon Web Services CloudTrail management event logs, CloudTrail S3 data event logs, EKS audit logs, DNS logs, Amazon EBS volume data, runtime activity belonging to container workloads, such as Amazon EKS, Amazon ECS (including Amazon Web Services Fargate), and Amazon EC2 instances. It uses threat intelligence feeds, such as lists of malicious IPs and domains, and machine learning to identify unexpected, potentially unauthorized, and malicious activity within your Amazon Web Services environment. This can include issues like escalations of privileges, uses of exposed credentials, or communication with malicious IPs, domains, or presence of malware on your Amazon EC2 instances and container workloads. For example, GuardDuty can detect compromised EC2 instances and container workloads serving malware, or mining bitcoin.

GuardDuty also monitors Amazon Web Services account access behavior for signs of compromise, such as unauthorized infrastructure deployments like EC2 instances deployed in a Region that has never been used, or unusual API calls like a password policy change to reduce password strength.

GuardDuty informs you about the status of your Amazon Web Services environment by producing security findings that you can view in the GuardDuty console or through Amazon EventBridge. For more information, see the Amazon GuardDuty User Guide.

Usage

```r
GuardDuty(config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

credentials Optional credentials shorthand for the config parameter

• **creds**:
  – **access_key_id**: AWS access key ID
  – **secret_access_key**: AWS secret access key
  – **session_token**: AWS temporary session token
• **profile**: The name of a profile to use. If not given, then the default profile is used.
• **anonymous**: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- guardduty(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
```
Operations

- `accept_administrator_invitation` Accepts the invitation to be a member account and get monitored by a GuardDuty administrator account.
- `accept_invitation` Accepts the invitation to be monitored by a GuardDuty administrator account.
- `archive_findings` Archives GuardDuty findings that are specified by the list of finding IDs.
- `create_detector` Creates a single GuardDuty detector.
- `create_filter` Creates a filter using the specified finding criteria.
- `create_ip_set` Creates a new IPSet, which is called a trusted IP list in the console user interface.
- `create_members` Creates member accounts of the current Amazon Web Services account by specifying a list of Amazon Web Services account IDs.
- `create_publishing_destination` Creates a publishing destination to export findings to.
- `create_sample_findings` Generates sample findings of types specified by the list of finding types.
- `create_threat_intel_set` Creates a new ThreatIntelSet.
- `decline_invitations` Declines invitations sent to the current member account by Amazon Web Services accounts specified by their account IDs.
- `delete_detector` Deletes an Amazon GuardDuty detector that is specified by the detector ID.
- `delete_filter` Deletes the filter specified by the filter name.
- `delete_ip_set` Deletes the IPSet specified by the ipSetId.
- `delete_invitations` Deletes invitations sent to the current member account by Amazon Web Services accounts specified by their account IDs.
- `delete_members` Deletes GuardDuty member accounts (to the current GuardDuty administrator account) specified by the account IDs.
- `delete_publishing_destination` Deletes the publishing definition with the specified destinationId.
- `delete_threat_intel_set` Deletes the ThreatIntelSet specified by the ThreatIntelSet ID.
- `describe_malware_scans` Returns a list of malware scans.
- `describe_organization_configuration` Returns information about the account selected as the delegated administrator for GuardDuty.
- `describe_publishing_destination` Returns information about the publishing destination specified by the provided destinationId.
- `disable_organization_admin_account` Removes the existing GuardDuty delegated administrator of the organization.
- `disassociate_from_administrator_account` Disassociates the current GuardDuty member account from its administrator account.
- `disassociate_from_master_account` Disassociates the current GuardDuty member account from its administrator account.
- `disassociate_members` Disassociates GuardDuty member accounts (from the current administrator account) specified by the account IDs.
- `enable_organization_admin_account` Designates an Amazon Web Services account within the organization as your GuardDuty delegated administrator.
- `get_administrator_account` Provides the details of the GuardDuty administrator account associated with the current member account.
- `get_coverage_statistics` Retrieves aggregated statistics for your account.
- `get_detector` Retrieves an Amazon GuardDuty detector specified by the detectorId.
- `get_filter` Retrieves the details of the filter specified by the filter name.
- `get_findings` Describes Amazon GuardDuty findings specified by finding IDs.
- `get_findings_statistics` Lists Amazon GuardDuty findings statistics for the specified detector ID.
- `get_invitations_count` Retrieves the count of all GuardDuty membership invitations that were sent to the current member account.
- `get_ip_set` Retrieves the IPSet specified by the ipSetId.
- `get_malware_scan_settings` Provides the details for the GuardDuty administrator account associated with the current member account.
- `get_organization_statistics` Provides aggregated statistics for your account.
- `get_remaining_free_trial_days` Provides the number of days left for each data source used in the free trial period.
get_threat_intel_set Retrieves the ThreatIntelSet that is specified by the ThreatIntelSet ID
get_usage_statistics Lists Amazon GuardDuty usage statistics over the last 30 days for the specified detector ID
invite_members Invites Amazon Web Services accounts to become members of an organization administered by the current Amazon Web Services account
list_coverage Lists coverage details for your GuardDuty account
list_detectors Lists detectorIds of all the existing Amazon GuardDuty detector resources
list_filters Returns a paginated list of the current filters
list_findings Lists GuardDuty findings for the specified detector ID
list_invitations Lists all GuardDuty membership invitations that were sent to the current Amazon Web Services account
list_ip_sets Lists the IPSets of the GuardDuty service specified by the detector ID
list_members Lists details about all member accounts for the current GuardDuty administrator account
list_organization_admin_accounts Lists the accounts designated as GuardDuty delegated administrators
list_publishing_destinations Returns a list of publishing destinations associated with the specified detectorId
list_threat_intel_sets Lists the ThreatIntelSets of the GuardDuty service specified by the detector ID
start_malware_scan Initiates the malware scan
start_monitoring_members Turns on GuardDuty monitoring of the specified member accounts
stop_monitoring_members Stops GuardDuty monitoring for the specified member accounts
tag_resource Adds tags to a resource
unarchive_findings Unarchives GuardDuty findings specified by the findingIds
untag_resource Removes tags from a resource
update_detector Updates the GuardDuty detector specified by the detector ID
update_filter Updates the filter specified by the filter name
update_findings_feedback Marks the specified GuardDuty findings as useful or not useful
update_ip_set Updates the IPSet specified by the IPSet ID
update_malware_scan_settings Updates the malware scan settings
update_member_detectors Contains information on member accounts to be updated
update_organization_configuration Configures the delegated administrator account with the provided values
update_publishing_destination Updates information about the publishing destination specified by the destinationId
update_threat_intel_set Updates the ThreatIntelSet specified by the ThreatIntelSet ID

Examples

```r
## Not run:
svc <- guardduty()
svc$accept_administrator_invitation(
  Foo = 123
)

## End(Not run)
```
Description

Health

The Health API provides access to the Health information that appears in the Health Dashboard. You can use the API operations to get information about events that might affect your Amazon Web Services and resources.

You must have a Business, Enterprise On-Ramp, or Enterprise Support plan from Amazon Web Services Support to use the Health API. If you call the Health API from an Amazon Web Services account that doesn’t have a Business, Enterprise On-Ramp, or Enterprise Support plan, you receive a SubscriptionRequiredException error.

For API access, you need an access key ID and a secret access key. Use temporary credentials instead of long-term access keys when possible. Temporary credentials include an access key ID, a secret access key, and a security token that indicates when the credentials expire. For more information, see Best practices for managing Amazon Web Services access keys in the Amazon Web Services General Reference.

You can use the Health endpoint health.us-east-1.amazonaws.com (HTTPS) to call the Health API operations. Health supports a multi-Region application architecture and has two regional endpoints in an active-passive configuration. You can use the high availability endpoint example to determine which Amazon Web Services Region is active, so that you can get the latest information from the API. For more information, see Accessing the Health API in the Health User Guide.

For authentication of requests, Health uses the Signature Version 4 Signing Process.

If your Amazon Web Services account is part of Organizations, you can use the Health organizational view feature. This feature provides a centralized view of Health events across all accounts in your organization. You can aggregate Health events in real time to identify accounts in your organization that are affected by an operational event or get notified of security vulnerabilities. Use the organizational view API operations to enable this feature and return event information. For more information, see Aggregating Health events in the Health User Guide.

When you use the Health API operations to return Health events, see the following recommendations:

- Use the eventScopeCode parameter to specify whether to return Health events that are public or account-specific.
- Use pagination to view all events from the response. For example, if you call the describe_events_for_organization operation to get all events in your organization, you might receive several page results. Specify the nextToken in the next request to return more results.

Usage

health(config = list(), credentials = list(), endpoint = NULL, region = NULL)

Arguments

config Optional configuration of credentials, endpoint, and/or region.

- credentials:

  - creds:
    - access_key_id: AWS access key ID
• secret_access_key: AWS secret access key
• session_token: AWS temporary session token
  – profile: The name of a profile to use. If not given, then the default
    profile is used.
  – anonymous: Set anonymous credentials.
• endpoint: The complete URL to use for the constructed client.
• region: The AWS Region used in instantiating the client.
• close_connection: Immediately close all HTTP connections.
• timeout: The time in seconds till a timeout exception is thrown when at-
  tempting to make a connection. The default is 60 seconds.
• s3_force_path_style: Set this to true to force the request to use path-style
  addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• sts_regional_endpoint: Set sts regional endpoint resolver to regional or

credentials Optional credentials shorthand for the config parameter
• creds:
  – access_key_id: AWS access key ID
  – secret_access_key: AWS secret access key
  – session_token: AWS temporary session token
• profile: The name of a profile to use. If not given, then the default profile
  is used.
• anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.
region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...) ,
where svc is the name you’ve assigned to the client. The available operations are listed in the Op-
erations section.

Service syntax

svc <- health(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = logical
  ),
  endpoint = "string",
  region = "string"
)
region = "string",
close_connection = "logical",
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
  creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
  ),
  profile = "string",
  anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

describe_affected_accounts_for_organization
describe_affected_entities
describe_event_aggregates
describe_event_aggregates_for_organization
describe_entity_aggregates
describe_entity_aggregates_for_organization
describe_event_details
describe_event_details_for_organization
describe_events
describe_events_for_organization
describe_event_types
describe_health_service_status_for_organization
disable_health_service_access_for_organization
enable_health_service_access_for_organization

Examples

## Not run:
svc <- health()
svc$describe_affected_accounts_for_organization(
  Foo = 123
)

## End(Not run)
**Description**

AWS HealthLake is a HIPAA eligible service that allows customers to store, transform, query, and analyze their FHIR-formatted data in a consistent fashion in the cloud.

**Usage**

```r
healthlake(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

**Arguments**

- `config`: Optional configuration of credentials, endpoint, and/or region.
  - `credentials`: Optional credentials shorthand for the `config` parameter
    - `creds`:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
    - `anonymous`: Set anonymous credentials.
    - `endpoint`: The complete URL to use for the constructed client.
    - `region`: The AWS Region used in instantiating the client.
    - `close_connection`: Immediately close all HTTP connections.
    - `timeout`: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
    - `s3_force_path_style`: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.

- `credentials`: Optional credentials shorthand for the `config` parameter
  - `creds`:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
csvc <- healthlake(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

**Operations**

- `create_fhir_datastore` Creates a data store that can ingest and export FHIR formatted data
IAM (AWS Identity and Access Management) is a web service for securely controlling access to Amazon Web Services services. With IAM, you can centrally manage users, security credentials such as access keys, and permissions that control which Amazon Web Services resources users and applications can access. For more information about IAM, see Identity and Access Management (IAM) and the Identity and Access Management User Guide.

Usage

```
iam(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

- **config**
  - Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:

Examples

```r
## Not run:
svc <- healthlake()
svc$create_fhir_datastore(
  Foo = 123
)
## End(Not run)
```
* access_key_id: AWS access key ID
* secret_access_key: AWS secret access key
* session_token: AWS temporary session token

- profile: The name of a profile to use. If not given, then the default profile is used.
- anonymous: Set anonymous credentials.

- endpoint: The complete URL to use for the constructed client.
- region: The AWS Region used in instantiating the client.
- close_connection: Immediately close all HTTP connections.
- timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
- sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter

- creds:
  - access_key_id: AWS access key ID
  - secret_access_key: AWS secret access key
  - session_token: AWS temporary session token

- profile: The name of a profile to use. If not given, then the default profile is used.
- anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- iam(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
```
Operations

add_client_id_to_open_id_connect_provider
add_role_to_instance_profile
add_user_to_group
attach_group_policy
attach_role_policy
attach_user_policy
change_password
create_access_key
create_account_alias
create_group
create_instance_profile
create_login_profile
create_open_id_connect_provider
create_policy
create_policy_version
create_role
create_saml_provider
create_service_linked_role
create_service_specific_credential
create_user
create_virtual_mfa_device
deactivate_mfa_device
delete_access_key
delete_account_alias
delete_account_password_policy

Adds a new client ID (also known as audience) to the list of client IDs associated with the specified IAM OpenID Connect (OIDC) provider resource.

Adds the specified IAM role to the specified instance profile.

Adds the specified user to the specified group.

Attaches the specified managed policy to the specified IAM group.

Attaches the specified managed policy to the specified IAM role.

Attaches the specified managed policy to the specified user.

Changes the password of the IAM user who is calling this operation.

Creates a new Amazon Web Services secret access key and corresponding Amazon Web Services access key ID for the specified user.

Creates an alias for your Amazon Web Services account.

Creates a new group.

Creates a new instance profile.

Creates a password for the specified IAM user.

Creates an IAM entity to describe an identity provider (IdP) that supports OpenID Connect (OIDC).

Creates a new managed policy for your Amazon Web Services account.

Creates a new version of the specified managed policy.

Creates a new role for your Amazon Web Services account.

Creates an IAM resource that describes an identity provider (IdP) that supports SAML 2.

Creates an IAM role that is linked to a specific Amazon Web Services service.

Generates a set of credentials consisting of a user name and password that can be used to access the specified service.

Creates a new IAM user for your Amazon Web Services account.

Creates a new virtual MFA device for the Amazon Web Services account.

Deactivates the specified MFA device and removes it from association with the specified IAM user.

Deletes the access key pair associated with the specified IAM user.

Deletes the account alias associated with the specified Amazon Web Services account.

Deletes the password policy for the Amazon Web Services account.
delete_group
delete_group_policy
delete_instance_profile
delete_login_profile
delete_open_id_connect_provider
delete_policy
delete_policy_version
delete_role
delete_role_permissions_boundary
delete_role_policy
delete_saml_provider
delete_server_certificate
delete_service_linked_role
delete_service_specific_credential
delete_signing_certificate
delete_ssh_public_key
delete_user
delete_user_permissions_boundary
delete_user_policy
delete_virtual_mfa_device
detach_group_policy
detach_role_policy
detach_user_policy
enable_mfa_device
generate_credential_report
generate_organizations_access_report
generate_service_last_accessed_details
get_access_key_last_used
get_account_authorization_details
get_account_password_policy
get_account_summary
get_context_keys_for_custom_policy
get_context_keys_for_principal_policy
generate_credential_report
group
group_policy
get_instance_profile
get_login_profile
generate_credential_report
get_mfa_device
get_open_id_connect_provider
get_account_authorization_details
get_account_password_policy
get_account_summary
get_context_keys_for_custom_policy
get_context_keys_for_principal_policy
generate_credential_report
get_group
get_group_policy
get_instance_profile
get_login_profile
get_mfa_device
get_open_id_connect_provider
get_organizations_access_report
get_policy
generate_credential_report
get_policy_version
get_role
get_role_policy
get_saml_provider
generate_credential_report
get_server_certificate
get_service_last_accessed_details

Deletes the specified IAM group
Deletes the specified inline policy that is embedded in the specified IAM group
Deletes the specified managed policy
Deletes the specified version from the specified managed policy
Deletes the specified role
Deletes the permissions boundary for the specified IAM role
Deletes the specified inline policy that is embedded in the specified IAM role
Deletes a SAML provider resource in IAM
Deletes the specified server certificate
Submits a service-linked role deletion request and returns a DeletionTaskId, which you can use to check the status of the deletion
Deletes the specified service-specific credential
Deletes a signing certificate associated with the specified IAM user
Deletes the specified SSH public key
Deletes the specified IAM user
Deletes the permissions boundary for the specified IAM user
Deletes the specified inline policy that is embedded in the specified IAM user
Deletes a virtual MFA device
Removes the specified managed policy from the specified IAM group
Removes the specified managed policy from the specified role
Removes the specified managed policy from the specified user
Enables the specified MFA device and associates it with the specified IAM user
Generates a credential report for the Amazon Web Services account
Generates a report for service last accessed data for Organizations
Gets a list of all of the context keys referenced in the input policies
Gets a list of all of the context keys referenced in all the IAM policies that the specified IAM user is associated with
Retrieves a credential report for the Amazon Web Services account
Retrieves a list of IAM users that are in the specified IAM group
Retrieves the specified inline policy document that is embedded in the specified IAM group
Retrieves information about the specified instance profile, including the user name for the specified IAM user
Retrieves information about an MFA device for a specified user
Retrieves information about the specified OpenID Connect (OIDC) provider
Retrieves the service last accessed data report for Organizations that was created using the GenerateOrganizationAccessReport operation
Retrieves information about the specified managed policy, including the role’s path, the role’s policy, and the role’s permissions
Retrieves information about the specified version of the specified managed policy
Retrieves information about the specified role, including the role’s path, the role’s policy, and the role’s permissions
Retrieves the specified inline policy document that is embedded with the specified role
Retrieves the SAML provider metadocument that was uploaded when the service was created
Retrieves information about the specified server certificate stored in IAM
Retrieves a service last accessed report that was created using the GenerateServiceLastAccessedDetails operation
get_service_last_accessed_details_with_entities
get_service_linked_role_deletion_status
get_ssh_public_key
get_user
get_user_policy
list_access_keys
list_account_aliases
list_attached_group_policies
list_attached_role_policies
list_attached_user_policies
list_entities_for_policy
list_group_policies
list_groups
list_groups_for_user
list_instance_profiles
list_instance_profiles_for_role
list_instance_profile_tags
list_mfa_devices
list_mfa_device_tags
list_open_id_connect_providers
list_open_id_connect_provider_tags
list_policies
list_policies_granting_service_access
list_policy_tags
list_policy_versions
list_role_policies
list_roles
list_role_tags
list_saml_providers
list_saml_provider_tags
list_server_certificates
list_server_certificate_tags
list_service_specific_credentials
list_signing_certificates
list_ssh_public_keys
list_user_policies
list_users
list_user_tags
list_virtual_mfa_devices
put_group_policy
put_role_permissions_boundary
put_role_policy
put_user_permissions_boundary
put_user_policy
remove_client_id_from_open_id_connect_provider
remove_role_from_instance_profile
remove_user_from_group
reset_service_specific_credential

After you generate a group or policy report using the GenerateServiceLastAccessedDetails operation, you can use the JobId parameter in GetServiceLastAccessedDetailsWithEntities to retrieve detailed information about the most recently accessed services.

get_service_linked_role_deletion_status
Retrieves the status of your service-linked role deletion.

get_ssh_public_key
Retrieves the specified SSH public key, including metadata about the key.

get_user
Retrieves information about the specified IAM user, including the user’s access keys.

get_user_policy
Retrieves the specified inline policy document that is embedded in the specified user.

list_account_aliases
Lists the account alias associated with the Amazon Web Services account.

list_attached_group_policies
Lists all managed policies that are attached to the specified IAM group.

list_attached_role_policies
Lists all managed policies that are attached to the specified IAM role.

list_attached_user_policies
Lists all managed policies that are attached to the specified IAM user.

list_entities_for_policy
Lists all IAM users, groups, and roles that the specified managed policy is attached to.

list_group_policies
Lists the names of the inline policies that are embedded in the specified IAM group.

list_groups
Lists the IAM groups that have the specified path prefix.

list_groups_for_user
Lists the IAM groups that the specified IAM user belongs to.

list_instance_profiles
Lists the instance profiles that have the specified path prefix.

list_instance_profiles_for_role
Lists the instance profiles that have the specified associated IAM role.

list_instance_profile_tags
Lists the tags that are attached to the specified IAM instance profile.

list_mfa_devices
Lists the MFA devices for an IAM user.

list_mfa_device_tags
Lists the tags that are attached to the specified IAM virtual multi-factor authentication (MFA) device.

list_open_id_connect_providers
Lists information about the IAM OpenID Connect (OIDC) provider resource objects defined in the account.

list_open_id_connect_provider_tags
Lists the tags that are attached to the specified OpenID Connect (OIDC) provider.

list_policies
Lists all the managed policies that are available in your Amazon Web Services account, including your own customer-defined managed policies and all Amazon Web Services managed policies.

list_policies_granting_service_access
Retrieves a list of policies that the IAM identity (user, group, or role) can use to access each specified service.

list_policy_tags
Lists the tags that are attached to the specified IAM customer managed policy.

list_policy_versions
Lists information about the versions of the specified managed policy, including the version that is currently set as the policy’s default version.

list_role_policies
Lists the names of the inline policies that are embedded in the specified IAM role.

list_roles
Lists the IAM roles that have the specified path prefix.

list_role_tags
Lists the IAM roles that have the specified role.

list_saml_providers
Lists the SAML provider resource objects defined in IAM in the account.

list_saml_provider_tags
Lists the tags that are attached to the specified Security Assertion Markup Language (SAML) identity provider.

list_server_certificates
Lists the server certificates stored in IAM that have the specified path prefix.

list_server_certificate_tags
Lists the tags that are attached to the specified IAM server certificate.

list_service_specific_credentials
Returns information about the service-specific credentials associated with the specified IAM instance profile.

list_signing_certificates
Returns information about the signing certificates associated with the specified IAM user.

list_ssh_public_keys
Returns information about the SSH public keys associated with the specified IAM user.

list_user_policies
Lists the names of the inline policies embedded in the specified IAM user.

list_users
List the IAM users that have the specified path prefix.

list_user_tags
Lists the tags that are attached to the specified IAM user.

list_virtual_mfa_devices
Lists the virtual MFA devices defined in the Amazon Web Services account.

put_group_policy
Adds or updates an inline policy document that is embedded in the specified group.

put_role_permissions_boundary
Adds or updates the policy that is specified as the IAM role’s permissions boundary.

put_role_policy
Adds or updates an inline policy document that is embedded in the specified role.

put_user_permissions_boundary
Adds or updates the policy that is specified as the IAM user’s permissions boundary.

put_user_policy
Adds or updates an inline policy document that is embedded in the specified user.

remove_client_id_from_open_id_connect_provider
Removes the specified client ID (also known as audience) from the list of registered client IDs for the specified IAM OpenID Connect (OIDC) provider.

remove_role_from_instance_profile
Removes the specified IAM role from the specified Amazon EC2 instance profile.

remove_user_from_group
Removes the specified user from the specified group.

reset_service_specific_credential
Resets the password for a service-specific credential.
resync_mfa_device
set_default_policy_version
set_security_token_service_preferences
simulate_custom_policy
simulate_principal_policy
tag_instance_profile
tag_mfa_device
tag_open_id_connect_provider
tag_policy
tag_role
tag_saml_provider
tag_server_certificate
tag_user
untag_instance_profile
untag_mfa_device
untag_open_id_connect_provider
untag_policy
untag_role
untag_saml_provider
untag_server_certificate
untag_user
update_access_key
update_account_password_policy
update_assume_role_policy
update_group
update_login_profile
update_open_id_connect_provider_thumbprint
update_role
update_role_description
update_saml_provider
update_server_certificate
update_service_specific_credential
update_signing_certificate
update_ssh_public_key
update_user
upload_server_certificate
upload_signing_certificate
upload_ssh_public_key

Synchronizes the specified MFA device with its IAM resource object on Amazon Web Services servers.
Sets the specified version of the specified policy as the policy’s default (operative) version.
Sets the specified version of the global endpoint token as the token version used for the Amazon Web Services account.
Simulate how a set of IAM policies and optionally a resource-based policy works with a list of API operations and Amazon Web Services resources to determine the policies' effective permissions.
Simulate how a set of IAM policies attached to an IAM entity works with a list of API operations and Amazon Web Services resources to determine the policies' effective permissions.
Adds one or more tags to an IAM instance profile.
Adds one or more tags to an IAM virtual multi-factor authentication (MFA) device.
Adds one or more tags to an OpenID Connect (OIDC)-compatible identity provider.
Adds one or more tags to an IAM customer managed policy.
Adds one or more tags to an IAM role.
Adds one or more tags to a Security Assertion Markup Language (SAML) identity provider.
Adds one or more tags to an IAM server certificate.
Adds one or more tags to an IAM user.
Removes the specified tags from the IAM instance profile.
Removes the specified tags from the IAM virtual multi-factor authentication (MFA) device.
Removes the specified tags from the specified OpenID Connect (OIDC) provider.
Removes the specified tags from the customer managed policy.
Removes the specified tags from the role.
Removes the specified tags from the specified Security Assertion Markup Language (SAML) identity provider.
Removes the specified tags from the IAM server certificate.
Removes the specified tags from the user.
Changes the status of the specified access key from Active to Inactive, or vice versa.
Updates the password policy settings for the Amazon Web Services account.
Updates the policy that grants an IAM entity permission to assume a role.
Updates the name and/or the path of the specified IAM group.
Changes the password for the specified IAM user.
Replaces the existing list of server certificate thumbprints associated with a role.
Updates the description or maximum session duration setting of a role.
Use UpdateRole instead.
Updates the metadata document for an existing SAML provider resource.
Updates the name and/or the path of the specified server certificate stored in IAM.
Sets the status of a service-specific credential to Active or Inactive.
Changes the status of the specified user signing certificate from active to inactive.
Sets the status of an IAM user's SSH public key to active or inactive.
Updates the name and/or the path of the specified IAM user.
Uploads a server certificate entity for the Amazon Web Services account.
Uploads an X.509 certificate.
Uploads an SSH public key and associates it with the specified IAM user.

Examples

```r
## Not run:
svc <- iam()

# The following add-client-id-to-open-id-connect-provider command adds the
# client ID my-application-ID to the OIDC provider named
# server.example.com:
svc$add_client_id_to_open_id_connect_provider(
```
iamrolesanywhere

IAM Roles Anywhere

Description

Identity and Access Management Roles Anywhere provides a secure way for your workloads such as servers, containers, and applications that run outside of Amazon Web Services to obtain temporary Amazon Web Services credentials. Your workloads can use the same IAM policies and roles you have for native Amazon Web Services applications to access Amazon Web Services resources. Using IAM Roles Anywhere eliminates the need to manage long-term credentials for workloads running outside of Amazon Web Services.

To use IAM Roles Anywhere, your workloads must use X.509 certificates issued by their certificate authority (CA). You register the CA with IAM Roles Anywhere as a trust anchor to establish trust between your public key infrastructure (PKI) and IAM Roles Anywhere. If you don’t manage your own PKI system, you can use Private Certificate Authority to create a CA and then use that to establish trust with IAM Roles Anywhere.

This guide describes the IAM Roles Anywhere operations that you can call programmatically. For more information about IAM Roles Anywhere, see the IAM Roles Anywhere User Guide.

Usage

iamrolesanywhere(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

Optional configuration of credentials, endpoint, and/or region.

- **credentials:*
  - **creds:*
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.
• **endpoint**: The complete URL to use for the constructed client.
• **region**: The AWS Region used in instantiating the client.
• **close_connection**: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

**credentials**
Optional credentials shorthand for the config parameter
- **creds**:
  - `access_key_id`: AWS access key ID
  - `secret_access_key`: AWS secret access key
  - `session_token`: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**
A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**
```r
svc <- iamrolesanywhere(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  )
)```
iamrolesanywhere

credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

create_profile  Creates a profile, a list of the roles that Roles Anywhere service is trusted to assume
create_trust_anchor  Creates a trust anchor to establish trust between IAM Roles Anywhere and your certificate authority (CA)
delete_attribute_mapping  Delete an entry from the attribute mapping rules enforced by a given profile
delete_crl  Deletes a certificate revocation list (CRL)
delete_profile  Deletes a profile
delete_trust_anchor  Deletes a trust anchor
disable_crl  Disables a certificate revocation list (CRL)
disable_profile  Disables a profile
disable_trust_anchor  Disables a trust anchor
enable_crl  Enables a certificate revocation list (CRL)
enable_profile  Enables temporary credential requests for a profile
enable_trust_anchor  Enables a trust anchor
get_crl  Gets a certificate revocation list (CRL)
get_profile  Gets a profile
get_subject  Gets a subject, which associates a certificate identity with authentication attempts
get_trust_anchor  Gets a trust anchor
import_crl  Imports the certificate revocation list (CRL)
list_crls  Lists all certificate revocation lists (CRL) in the authenticated account and Amazon Web Services Region
list_profiles  Lists all profiles in the authenticated account and Amazon Web Services Region
list_subjects  Lists the subjects in the authenticated account and Amazon Web Services Region
list_tags_for_resource  Lists the tags attached to the resource
list_trust_anchors  Lists the trust anchors in the authenticated account and Amazon Web Services Region
put_attribute_mapping  Put an entry in the attribute mapping rules that will be enforced by a given profile
put_notification_settings  Attaches a list of notification settings to a trust anchor
reset_notification_settings  Resets the custom notification setting to IAM Roles Anywhere default setting
tag_resource  Attaches tags to a resource
untag_resource  Removes tags from the resource
update_crl  Updates the certificate revocation list (CRL)
update_profile  Updates a profile, a list of the roles that IAM Roles Anywhere service is trusted to assume
update_trust_anchor  Updates a trust anchor
Examples

## Not run:
svc <- iamrolesanywhere()
svc$create_profile(
  Foo = 123
)

## End(Not run)

identitystore  AWS SSO Identity Store

Description

The Identity Store service used by IAM Identity Center provides a single place to retrieve all of your identities (users and groups). For more information, see the IAM Identity Center User Guide. This reference guide describes the identity store operations that you can call programmatically and includes detailed information about data types and errors.

IAM Identity Center uses the sso and identitystore API namespaces.

Usage

identitystore(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)

Arguments

Optional configuration of credentials, endpoint, and/or region.

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter

creds:
  - access_key_id: AWS access key ID
  - secret_access_key: AWS secret access key
  - session_token: AWS temporary session token

profile: The name of a profile to use. If not given, then the default profile is used.

anonymous: Set anonymous credentials.

default

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```
svc <- identitystore(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  region = "string")
```
secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

create_group Creates a group within the specified identity store
create_group_membership Creates a relationship between a member and a group
create_user Creates a user within the specified identity store
delete_group Delete a group within an identity store given GroupId
delete_group_membership Delete a membership within a group given MembershipId
delete_user Deletes a user within an identity store given UserId
describe_group Retrieves the group metadata and attributes from GroupId in an identity store
describe_group_membership Retrieves membership metadata and attributes from MembershipId in an identity store
describe_user Retrieves the user metadata and attributes from the UserId in an identity store
get_group_id Retrieves GroupId in an identity store
get_group_membership_id Retrieves the MembershipId in an identity store
get_user_id Retrieves the UserId in an identity store
is_member_in_groups Checks the user’s membership in all requested groups and returns if the member exists
list_group_memberships For the specified group in the specified identity store, returns the list of all GroupMembership objects
list_group_memberships_for_member For the specified member in the specified identity store, returns the list of all GroupMembership objects
list_groups Lists all groups in the identity store
list_users Lists all users in the identity store
update_group For the specified group in the specified identity store, updates the group metadata and attributes
update_user For the specified user in the specified identity store, updates the user metadata and attributes

Examples

## Not run:
svc <- identitystore()
svc$create_group(
   Foo = 123
)

## End(Not run)
**Description**

EC2 Image Builder is a fully managed Amazon Web Services service that makes it easier to automate the creation, management, and deployment of customized, secure, and up-to-date "golden" server images that are pre-installed and pre-configured with software and settings to meet specific IT standards.

**Usage**

```python
def imagebuilder(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

**Arguments**

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **creds**:
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
    - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.
- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
- `session_token`: AWS temporary session token

- `profile`: The name of a profile to use. If not given, then the default profile is used.

- `anonymous`: Set anonymous credentials.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- imagebuilder(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

**Operations**
cancel_image_creation
CancelImageCreation cancels the creation of Image
cancel_lifecycle_execution
Cancel a specific image lifecycle policy runtime instance
create_component
Creates a new component that can be used to build, validate, test, and assess your image
create_container_recipe
Creates a new container recipe
create_distribution_configuration
Creates a new distribution configuration
create_image
Creates a new image
create_image_pipeline
Creates a new image pipeline
create_image_recipe
Creates a new image recipe
create_infrastructure_configuration
Creates a new infrastructure configuration
create_lifecycle_policy
Create a lifecycle policy resource
create_workflow
Create a new workflow or a new version of an existing workflow
delete_component
Deletes a component build version
delete_container_recipe
Deletes a container recipe
delete_distribution_configuration
Deletes a distribution configuration
delete_image
Deletes an Image Builder image resource
delete_image_pipeline
Deletes an image pipeline
delete_image_recipe
Deletes an image recipe
delete_infrastructure_configuration
Deletes an infrastructure configuration
delete_lifecycle_policy
Delete the specified lifecycle policy resource
delete_workflow
Deletes a specific workflow resource
get_component
Gets a component object
get_component_policy
Gets a component policy
get_container_recipe
Retrieves a container recipe
get_container_recipe_policy
Retrieves the policy for a container recipe
get_distribution_configuration
Gets a distribution configuration
get_image
Gets an image
get_image_pipeline
Gets an image pipeline
get_image_policy
Gets an image policy
get_image_recipe
Gets an image recipe
get_infrastructure_configuration
Gets an infrastructure configuration
get_lifecycle_execution
Get the runtime information that was logged for a specific runtime instance of the lifecycle policy
get_lifecycle_policy
Get details for the specified image lifecycle policy
get_workflow
Get a workflow resource object
get_workflow_execution
Get the runtime information that was logged for a specific runtime instance of the workflow
get_workflow_step_execution
Get the runtime information that was logged for a specific runtime instance of the workflow step
import_component
Imports a component and transforms its data into a component document
import_vm_image
When you export your virtual machine (VM) from its virtualization environment, that process creates a set of one or more disk container files that act as snapshots of your VM's environment, settings, and data
list_component_build_versions
Returns the list of component build versions for the specified semantic version
list_components
Returns the list of components that can be filtered by name, or by using the listed filters
list_container_recipes
Returns a list of container recipes
list_distribution_configurations
Returns a list of distribution configurations
list_image_build_versions
Returns a list of image build versions
list_image_packages
List the Packages that are associated with an Image Build Version, as determined by Amazon Web Services Systems Manager Inventory at build time
list_image_pipeline_images
Returns a list of images created by the specified pipeline
list_image_pipelines
Returns a list of image pipelines
list_image_recipes
Returns a list of image recipes
list_images
Returns the list of images that you have access to
<table>
<thead>
<tr>
<th>Function Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>list_image_scan_finding_aggregations</code></td>
<td>Returns a list of image scan aggregations for your account</td>
</tr>
<tr>
<td><code>list_image_scan_findings</code></td>
<td>Returns a list of image scan findings for your account</td>
</tr>
<tr>
<td><code>list_infrastructure_configurations</code></td>
<td>Returns a list of infrastructure configurations</td>
</tr>
<tr>
<td><code>list_lifecycle_execution_resources</code></td>
<td>List resources that the runtime instance of the image lifecycle identified for lifecycle actions</td>
</tr>
<tr>
<td><code>list_lifecycle_executions</code></td>
<td>Get the lifecycle runtime history for the specified resource</td>
</tr>
<tr>
<td><code>list_lifecycle_policies</code></td>
<td>Get a list of lifecycle policies in your Amazon Web Services account</td>
</tr>
<tr>
<td><code>list_tags_for_resource</code></td>
<td>Returns the list of tags for the specified resource</td>
</tr>
<tr>
<td><code>list_waiting_workflow_steps</code></td>
<td>Get a list of workflow steps that are waiting for action for workflows in your Amazon Web Services account</td>
</tr>
<tr>
<td><code>list_workflows</code></td>
<td>Returns a list of workflow runtime instance metadata objects for a specific workflow resource</td>
</tr>
<tr>
<td><code>list_workflows</code></td>
<td>Lists workflow build versions based on filtering parameters</td>
</tr>
<tr>
<td><code>list_workflow_step_executions</code></td>
<td>Returns runtime data for each step in a runtime instance of the workflow that you specified</td>
</tr>
<tr>
<td><code>put_component_policy</code></td>
<td>Applies a policy to a component</td>
</tr>
<tr>
<td><code>put_container_recipe_policy</code></td>
<td>Applies a policy to a container image</td>
</tr>
<tr>
<td><code>put_image_policy</code></td>
<td>Applies a policy to an image</td>
</tr>
<tr>
<td><code>put_image_recipe_policy</code></td>
<td>Applies a policy to an image recipe</td>
</tr>
<tr>
<td><code>send_workflow_step_action</code></td>
<td>Pauses or resumes image creation when the associated workflow runs a WaitForAction step</td>
</tr>
<tr>
<td><code>start_image_pipeline_execution</code></td>
<td>Manually triggers a pipeline to create an image</td>
</tr>
<tr>
<td><code>start_resource_state_update</code></td>
<td>Begin asynchronous resource state update for lifecycle changes to the specified image resources</td>
</tr>
<tr>
<td><code>tag_resource</code></td>
<td>Adds a tag to a resource</td>
</tr>
<tr>
<td><code>untag_resource</code></td>
<td>Removes a tag from a resource</td>
</tr>
<tr>
<td><code>update_distribution_configuration</code></td>
<td>Updates a new distribution configuration</td>
</tr>
<tr>
<td><code>update_image_pipeline</code></td>
<td>Updates an image pipeline</td>
</tr>
<tr>
<td><code>update_infrastructure_configuration</code></td>
<td>Updates a new infrastructure configuration</td>
</tr>
<tr>
<td><code>update_lifecycle_policy</code></td>
<td>Update the specified lifecycle policy</td>
</tr>
</tbody>
</table>

### Examples

```r
## Not run:
svc <- imagebuilder()
svc$cancel_image_creation(
  Foo = 123
)

## End(Not run)
```

---

**inspector**  

*Amazon Inspector*

---

**Description**

Amazon Inspector enables you to analyze the behavior of your AWS resources and to identify potential security issues. For more information, see [Amazon Inspector User Guide](#).
Usage

inspectors(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

config Optional configuration of credentials, endpoint, and/or region.

- **credentials**: 
  - **creds**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

- **endpoint**: The complete URL to use for the constructed client.

- **region**: The AWS Region used in instantiating the client.

- **close_connection**: Immediately close all HTTP connections.

- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

- **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter

- **creds**: 
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.

- **anonymous**: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```r
svc <- inspector(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- **add_attributes_to_findings**: Assigns attributes (key and value pairs) to the findings that are specified by the ARNs of the findings.
- **create_assessment_target**: Creates a new assessment target using the ARN of the resource group that is generated by `CreateResourceGroup`.
- **create_assessment_template**: Creates an assessment template for the assessment target that is specified by the ARN of the assessment target.
- **create_exclusions_preview**: Starts the generation of an exclusions preview for the specified assessment template.
- **create_resource_group**: Creates a resource group using the specified set of tags (key and value pairs) that are used to select the EC2 instances to be included in an Amazon Inspector assessment target.
- **delete_assessment_run**: Deletes the assessment run that is specified by the ARN of the assessment run.
- **delete_assessment_target**: Deletes the assessment target that is specified by the ARN of the assessment target.
- **delete_assessment_template**: Deletes the assessment template that is specified by the ARN of the assessment template.
- **describe_assessment_runs**: Describes the assessment runs that are specified by the ARNs of the assessment runs.
- **describe_assessment_targets**: Describes the assessment targets that are specified by the ARNs of the assessment targets.
- **describe_assessment_templates**: Describes the assessment templates that are specified by the ARNs of the assessment templates.
- **describe_cross_account_access_role**: Describes the IAM role that enables Amazon Inspector to access your AWS account.
- **describe_exclusions**: Describes the exclusions that are specified by the exclusions’ ARNs.
- **describe_findings**: Describes the findings that are specified by the ARNs of the findings.
describe_resource_groups
describe_rules_packages
get_assessment_report
get_exclusions_preview
get_telemetry_metadata
list_assessment_run_agents
list_assessment_runs
list_assessment_targets
list_assessment_templates
list_event_subscriptions
list_exclusions
list_findings
list_rules_packages
list_tags_for_resource
preview_agents
register_cross_account_access_role
remove_attributes_from_findings
set_tags_for_resource
start_assessment_run
stop_assessment_run
subscribe_to_event
unsubscribe_from_event
update_assessment_target

Describes the resource groups that are specified by the ARNs of the resource groups
Describes the rules packages that are specified by the ARNs of the rules packages
Produces an assessment report that includes detailed and comprehensive results of a specified assessment run
Retrieves the exclusions preview (a list of ExclusionPreview objects) specified by the preview token
Information about the data that is collected for the specified assessment run
Lists the agents of the assessment runs that are specified by the ARNs of the assessment runs
Lists the assessment runs that correspond to the assessment templates that are specified by the ARNs of the assessment targets within this AWS account
Lists the assessment templates that correspond to the assessment targets that are specified by the ARNs of the assessment targets
Lists all the event subscriptions for the assessment template that is specified by the ARN of the assessment template
Lists exclusions that are generated by the assessment run
Lists findings that are generated by the assessment runs that are specified by the ARNs of the assessment runs
Lists all available Amazon Inspector rules packages
Lists all tags associated with an assessment template
Previews the agents installed on the EC2 instances that are part of the specified assessment target
Registers the IAM role that grants Amazon Inspector access to AWS Services needed to perform security assessments
Removes entire attributes (key and value pairs) from the findings that are specified by the ARNs of the findings
Sets tags (key and value pairs) to the assessment template that is specified by the ARN of the assessment template
Starts the assessment run specified by the ARN of the assessment template
Stops the assessment run that is specified by the ARN of the assessment run
Enables the process of sending Amazon Simple Notification Service (SNS) notifications about specified events
Disables the process of sending Amazon Simple Notification Service (SNS) notifications about specified events
Updates the assessment target that is specified by the ARN of the assessment target

Examples

```r
## Not run:
svc <- inspector()
# Assigns attributes (key and value pairs) to the findings that are
# specified by the ARNs of the findings.
svc$add_attributes_to_findings(
  attributes = list(
    list(
      key = "Example",
      value = "example"
    )
  ),
  findingArns = list(
    "arn:aws:inspector:us-west-2:123456789012:target/0-0kFIPuq/template/0-..."
  ))
## End(Not run)
```
Amazon Inspector is a vulnerability discovery service that automates continuous scanning for security vulnerabilities within your Amazon EC2, Amazon ECR, and Amazon Web Services Lambda environments.

Usage

```python
inspector2(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**: Credentials
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
    - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends...

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**: Credentials
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
    - **session_token**: AWS temporary session token
• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**
A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**
```
svc <- inspector2(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

**Operations**

- **associate_member** Associates an Amazon Web Services account with an Amazon Inspector delegated administrator.
batch_get_account_status
batch_get_code_snippet
batch_get_finding_details
batch_get_free_trial_info
batch_get_member_ec_2_deep_inspection_status
batch_update_member_ec_2_deep_inspection_status
cancel_findings_report
cancel.sbom.export
create_cis_scan_configuration
create.filter
cancel_findings_report
cancel.sbom.export
delete_cis_scan_configuration
delete.filter
describe.organization_configuration
disable
disable.delegated.admin.account
disassociate.member
enable
enable.delegated.admin.account
get.cis.scan.report
get.cis.scan.result.details
get.configuration
get.delegated.admin.account
get.ec.2.deep.inspection.configuration
get.encryption.key
get.findings.report.status
get.member
get.sbom.export
list.account.permissions
list.cis.scan.configurations
list.cis.scan.results.aggregated.by.checks
list.cis.scan.results.aggregated.by.target.resource
list.cis.scans
list.coverage
list.coverage.statistics
list.delegated.admin.accounts
list.filters
list.finding.aggregations
list.findings
list.members
list.tags.for.resource
list.usage.totals
reset.encryption.key
search.vulnerabilities
send.cis.session.health
send.cis.session.telemetry
start.cis.session

Retrieves the Amazon Inspector status of multiple Amazon Web Services accounts.
Retrieves code snippets from findings that Amazon Inspector detected vulnerabilities in.
Gets vulnerability details for findings.
Gets free trial status for multiple Amazon Web Services accounts.
Retrieves Amazon Inspector deep inspection activation status of multiple member accounts.
Activates or deactivates Amazon Inspector deep inspection for the provided member accounts.
Cancels the given findings report.
Cancels a software bill of materials (SBOM) report.
Creates a CIS scan configuration.
Creates a filter resource using specified filter criteria.
Creates a finding report.
Creates a software bill of materials (SBOM) report.
deletes a CIS scan configuration.
deletes a filter resource.
device Amazon Inspector configuration settings for an Amazon Web Services organization.
Disables Amazon Inspector scans for one or more Amazon Web Services accounts.
Disables the Amazon Inspector delegated administrator for your organization.
Disassociates a member account from an Amazon Inspector delegated administrator.
Enables Amazon Inspector scans for one or more Amazon Web Services accounts.
Enables the Amazon Inspector delegated administrator for your organization.
Retrieves a CIS scan report.
Retrieves CIS scan result details.
Retrieves setting configurations for Inspector scans.
Retrieves information about the Amazon Inspector delegated administrator.
Retrieves the activation status of Amazon Inspector deep inspection for member accounts.
Gets an encryption key.
Gets the status of a findings report.
Gets details of a software bill of materials (SBOM) report.
Lists the permissions an account has to configure Amazon Inspector.
Lists CIS scan configurations.
Lists scan results aggregated by checks.
Lists scan results aggregated by a target resource.
Returns a CIS scan list.
Lists coverage details for your environment.
Lists Amazon Inspector coverage statistics for your environment.
Lists information about the Amazon Inspector delegated administrator.
Lists the filters associated with your account.
Lists aggregated finding data for your environment based on specific criteria.
Lists findings for your environment.
Lists members associated with the Amazon Inspector delegated administrator.
Lists all tags attached to a given resource.
Lists the Amazon Inspector usage totals over the last 30 days.
Resets an encryption key.
Lists Amazon Inspector coverage details for a specific vulnerability.
Sends a CIS session health.
Sends a CIS session telemetry.
Starts a CIS session.
stop_cis_session  
tag_resource  
untag_resource  
update_cis_scan_configuration  
update_configuration  
update_ec_2_deep_inspection_configuration  
update_encryption_key  
update_filter  
update_organization_configuration  
update_org_ec_2_deep_inspection_configuration

Examples

```r
## Not run:
svc <- inspector2()
svc$associate_member(
  Foo = 123
)
## End(Not run)
```

ivs  
Amazon Interactive Video Service

Description

Introduction

The Amazon Interactive Video Service (IVS) API is REST compatible, using a standard HTTP API and an Amazon Web Services EventBridge event stream for responses. JSON is used for both requests and responses, including errors.

The API is an Amazon Web Services regional service. For a list of supported regions and Amazon IVS HTTPS service endpoints, see the Amazon IVS page in the Amazon Web Services General Reference.

*All API request parameters and URLs are case sensitive.*

For a summary of notable documentation changes in each release, see Document History.

Allowed Header Values

- Accept: application/json
- Accept-Encoding: gzip, deflate
- Content-Type: application/json

Key Concepts
• **Channel** — Stores configuration data related to your live stream. You first create a channel and then use the channel’s stream key to start your live stream.

• **Stream key** — An identifier assigned by Amazon IVS when you create a channel, which is then used to authorize streaming. *Treat the stream key like a secret, since it allows anyone to stream to the channel.*

• **Playback key pair** — Video playback may be restricted using playback-authorization tokens, which use public-key encryption. A playback key pair is the public-private pair of keys used to sign and validate the playback-authorization token.

• **Recording configuration** — Stores configuration related to recording a live stream and where to store the recorded content. Multiple channels can reference the same recording configuration.

• **Playback restriction policy** — Restricts playback by countries and/or origin sites.

For more information about your IVS live stream, also see Getting Started with IVS Low-Latency Streaming.

**Tagging**

A *tag* is a metadata label that you assign to an Amazon Web Services resource. A tag comprises a *key* and a *value*, both set by you. For example, you might set a tag as `topic:nature` to label a particular video category. See Tagging Amazon Web Services Resources for more information, including restrictions that apply to tags and "Tag naming limits and requirements"; Amazon IVS has no service-specific constraints beyond what is documented there.

Tags can help you identify and organize your Amazon Web Services resources. For example, you can use the same tag for different resources to indicate that they are related. You can also use tags to manage access (see Access Tags).

The Amazon IVS API has these tag-related endpoints: `tag_resource`, `untag_resource`, and `list_tags_for_resource`. The following resources support tagging: Channels, Stream Keys, Playback Key Pairs, and Recording Configurations.

At most 50 tags can be applied to a resource.

**Authentication versus Authorization**

Note the differences between these concepts:

• **Authentication** is about verifying identity. You need to be authenticated to sign Amazon IVS API requests.

• **Authorization** is about granting permissions. Your IAM roles need to have permissions for Amazon IVS API requests. In addition, authorization is needed to view Amazon IVS private channels. (Private channels are channels that are enabled for "playback authorization.")

**Authentication**

All Amazon IVS API requests must be authenticated with a signature. The Amazon Web Services Command-Line Interface (CLI) and Amazon IVS Player SDKs take care of signing the underlying API calls for you. However, if your application calls the Amazon IVS API directly, it’s your responsibility to sign the requests.

You generate a signature using valid Amazon Web Services credentials that have permission to perform the requested action. For example, you must sign PutMetadata requests with a signature generated from a user account that has the `ivs:PutMetadata` permission.

For more information:
• Authentication and generating signatures — See Authenticating Requests (Amazon Web Services Signature Version 4) in the Amazon Web Services General Reference.


Amazon Resource Names (ARNs)
ARNs uniquely identify AWS resources. An ARN is required when you need to specify a resource unambiguously across all of AWS, such as in IAM policies and API calls. For more information, see Amazon Resource Names in the AWS General Reference.

Channel Endpoints
• create_channel — Creates a new channel and an associated stream key to start streaming.
• get_channel — Gets the channel configuration for the specified channel ARN.
• batch_get_channel — Performs get_channel on multiple ARNs simultaneously.
• list_channels — Gets summary information about all channels in your account, in the Amazon Web Services region where the API request is processed. This list can be filtered to match a specified name or recording-configuration ARN. Filters are mutually exclusive and cannot be used together. If you try to use both filters, you will get an error (409 Conflict Exception).
• update_channel — Updates a channel's configuration. This does not affect an ongoing stream of this channel. You must stop and restart the stream for the changes to take effect.
• delete_channel — Deletes the specified channel.

Playback Restriction Policy Endpoints
• create_playback_restriction_policy — Creates a new playback restriction policy, for constraining playback by countries and/or origins.
• delete_playback_restriction_policy — Deletes the specified playback restriction policy.
• get_playback_restriction_policy — Gets the specified playback restriction policy.
• list_playback_restriction_policies — Gets summary information about playback restriction policies.
• update_playback_restriction_policy — Updates a specified playback restriction policy.

Private Channel Endpoints
For more information, see Setting Up Private Channels in the Amazon IVS User Guide.

• import_playback_key_pair — Imports the public portion of a new key pair and returns its arn and fingerprint. The privateKey can then be used to generate viewer authorization tokens, to grant viewers access to private channels (channels enabled for playback authorization).
• get_playback_key_pair — Gets a specified playback authorization key pair and returns the arn and fingerprint. The privateKey held by the caller can be used to generate viewer authorization tokens, to grant viewers access to private channels.
• list_playback_key_pairs — Gets summary information about playback key pairs.
• delete_playback_key_pair — Deletes a specified authorization key pair. This invalidates future viewer tokens generated using the key pair's privateKey.
• `start_viewer_session_revocation` — Starts the process of revoking the viewer session associated with a specified channel ARN and viewer ID. Optionally, you can provide a version to revoke viewer sessions less than and including that version.

• `batch_start_viewer_session_revocation` — Performs `start_viewer_session_revocation` on multiple channel ARN and viewer ID pairs simultaneously.

Recording Configuration Endpoints

• `create_recording_configuration` — Creates a new recording configuration, used to enable recording to Amazon S3.

• `get_recording_configuration` — Gets the recording-configuration metadata for the specified ARN.

• `list_recording_configurations` — Gets summary information about all recording configurations in your account, in the Amazon Web Services region where the API request is processed.

• `delete_recording_configuration` — Deletes the recording configuration for the specified ARN.

Stream Endpoints

• `get_stream` — Gets information about the active (live) stream on a specified channel.

• `get_stream_session` — Gets metadata on a specified stream.

• `list_streams` — Gets summary information about live streams in your account, in the Amazon Web Services region where the API request is processed.

• `list_stream_sessions` — Gets a summary of current and previous streams for a specified channel in your account, in the AWS region where the API request is processed.

• `stop_stream` — Disconnects the incoming RTMPS stream for the specified channel. Can be used in conjunction with `delete_stream_key` to prevent further streaming to a channel.

• `put_metadata` — Inserts metadata into the active stream of the specified channel. At most 5 requests per second per channel are allowed, each with a maximum 1 KB payload. (If 5 TPS is not sufficient for your needs, we recommend batching your data into a single `PutMetadata` call.) At most 155 requests per second per account are allowed.

Stream Key Endpoints

• `create_stream_key` — Creates a stream key, used to initiate a stream, for the specified channel ARN.

• `get_stream_key` — Gets stream key information for the specified ARN.

• `batch_get_stream_key` — Performs `get_stream_key` on multiple ARNs simultaneously.

• `list_stream_keys` — Gets summary information about stream keys for the specified channel.

• `delete_stream_key` — Deletes the stream key for the specified ARN, so it can no longer be used to stream.

Amazon Web Services Tags Endpoints

• `tag_resource` — Adds or updates tags for the Amazon Web Services resource with the specified ARN.
• `untag_resource` — Removes tags from the resource with the specified ARN.

• `list_tags_for_resource` — Gets information about Amazon Web Services tags for the specified ARN.

Usage

```r
ivs(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

- **config** Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - `creds`:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
    - `anonymous`: Set anonymous credentials.
  - `endpoint`: The complete URL to use for the constructed client.
  - `region`: The AWS Region used in instantiating the client.
  - `close_connection`: Immediately close all HTTP connections.
  - `timeout`: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - `s3_force_path_style`: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.

- **credentials** Optional credentials shorthand for the config parameter
  - `creds`:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
    - `anonymous`: Set anonymous credentials.

- **endpoint** Optional shorthand for complete URL to use for the constructed client.

- **region** Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...),` where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

svc <- ivs(
  config = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string",
  close_connection = "logical",
  timeout = "numeric",
  s3_force_path_style = "logical",
  sts_regional_endpoint = "string"
),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)

Operations

- batch_get_channel: Performs GetChannel on multiple ARNs simultaneously
- batch_get_stream_key: Performs GetStreamKey on multiple ARNs simultaneously
- batch_start_viewer_session_revocation: Performs StartViewerSessionRevocation on multiple channel ARN and viewer ID pairs
- create_channel: Creates a new channel and an associated stream key to start streaming
- create_playback_restriction_policy: Creates a new playback restriction policy, for constraining playback by countries and/or origins
- create_recording_configuration: Creates a new recording configuration, used to enable recording to Amazon S3
- create_stream_key: Creates a stream key, used to initiate a stream, for the specified channel ARN
- delete_channel: Deletes the specified channel and its associated stream keys
- delete_playback_key_pair: Deletes the specified playback key pair
- delete_playback_restriction_policy: Deletes the specified playback restriction policy
- delete_recording_configuration: Deletes the recording configuration for the specified ARN
- delete_stream_key: Deletes the stream key for the specified ARN, so it can no longer be used to stream
- get_channel: Gets the channel configuration for the specified channel ARN
- get_playback_key_pair: Gets a specified playback authorization key pair and returns the arn and fingerprint
get_playback_restriction_policy: Gets the specified playback restriction policy.
get_recording_configuration: Gets the recording configuration for the specified ARN.
get_stream: Gets information about the active (live) stream on a specified channel.
get_stream_key: Gets stream-key information for a specified ARN.
get_stream_session: Gets metadata on a specified stream.
import_playback_key_pair: Imports the public portion of a new key pair and returns its ARN and fingerprint.
list_channels: Gets summary information about all channels in your account, in the Amazon Web Services region where the API request is processed.
list_playback_key_pairs: Gets summary information about playback key pairs.
list_playback_restriction_policies: Gets summary information about playback restriction policies.
list_recording_configurations: Gets summary information about all recording configurations in your account, in the Amazon Web Services region where the API request is processed.
list_stream_keys: Gets summary information about stream keys for the specified channel.
list_streams: Gets summary information about live streams in your account, in the Amazon Web Services region where the API request is processed.
list_stream_sessions: Gets a summary of current and previous streams for a specified channel in your account.
list_tags_for_resource: Gets information about Amazon Web Services tags for the specified ARN.
put_metadata: Inserts metadata into the active stream of the specified channel.
start_viewer_session_revocation: Starts the process of revoking the viewer session associated with a specified channel ARN and viewer ID.
stop_stream: Disconnects the incoming RTMPS stream for the specified channel.
tag_resource: Adds or updates tags for the Amazon Web Services resource with the specified ARN.
tag_resource: Removes tags from the Amazon Web Services resource with the specified ARN.
update_channel: Updates a channel’s configuration.
update_playback_restriction_policy: Updates a specified playback restriction policy.

Examples

```r
## Not run:
svc <- ivs()
svc$batch_get_channel(
  Foo = 123
)
## End(Not run)
```

ivschat  Amazon Interactive Video Service Chat

Description

Introduction

The Amazon IVS Chat control-plane API enables you to create and manage Amazon IVS Chat resources. You also need to integrate with the Amazon IVS Chat Messaging API, to enable users to interact with chat rooms in real time.

The API is an AWS regional service. For a list of supported regions and Amazon IVS Chat HTTPS service endpoints, see the Amazon IVS Chat information on the Amazon IVS page in the AWS General Reference.

Notes on terminology:
• You create service applications using the Amazon IVS Chat API. We refer to these as applications.
• You create front-end client applications (browser and Android/iOS apps) using the Amazon IVS Chat Messaging API. We refer to these as clients.

Key Concepts

• **LoggingConfiguration** — A configuration that allows customers to store and record sent messages in a chat room.
• **Room** — The central Amazon IVS Chat resource through which clients connect to and exchange chat messages.

Tagging

A *tag* is a metadata label that you assign to an AWS resource. A tag comprises a *key* and a *value*, both set by you. For example, you might set a tag as `topic:nature` to label a particular video category. See [Tagging AWS Resources](#) for more information, including restrictions that apply to tags and “Tag naming limits and requirements”: Amazon IVS Chat has no service-specific constraints beyond what is documented there.

Tags can help you identify and organize your AWS resources. For example, you can use the same tag for different resources to indicate that they are related. You can also use tags to manage access (see [Access Tags](#)).

The Amazon IVS Chat API has these tag-related endpoints: `tag_resource`, `untag_resource`, and `list_tags_for_resource`. The following resource supports tagging: `Room`.

At most 50 tags can be applied to a resource.

API Access Security

Your Amazon IVS Chat applications (service applications and clients) must be authenticated and authorized to access Amazon IVS Chat resources. Note the differences between these concepts:

• **Authentication** is about verifying identity. Requests to the Amazon IVS Chat API must be signed to verify your identity.
• **Authorization** is about granting permissions. Your IAM roles need to have permissions for Amazon IVS Chat API requests.

Users (viewers) connect to a room using secure access tokens that you create using the `create_chat_token` endpoint through the AWS SDK. You call `CreateChatToken` for every user’s chat session, passing identity and authorization information about the user.

Signing API Requests

HTTP API requests must be signed with an AWS SigV4 signature using your AWS security credentials. The AWS Command Line Interface (CLI) and the AWS SDKs take care of signing the underlying API calls for you. However, if your application calls the Amazon IVS Chat HTTP API directly, it’s your responsibility to sign the requests.

You generate a signature using valid AWS credentials for an IAM role that has permission to perform the requested action. For example, `DeleteMessage` requests must be made using an IAM role that has the `ivschat:DeleteMessage` permission.

For more information:
• Authentication and generating signatures — See Authenticating Requests (Amazon Web Services Signature Version 4) in the Amazon Web Services General Reference.


Amazon Resource Names (ARNs)
ARNs uniquely identify AWS resources. An ARN is required when you need to specify a resource unambiguously across all of AWS, such as in IAM policies and API calls. For more information, see Amazon Resource Names in the AWS General Reference.

Messaging Endpoints
• delete_message — Sends an event to a specific room which directs clients to delete a specific message: that is, unrender it from view and delete it from the client’s chat history. This event’s EventName is aws:DELETE_MESSAGE. This replicates the DeleteMessage WebSocket operation in the Amazon IVS Chat Messaging API.

• disconnect_user — Disconnects all connections using a specified user ID from a room. This replicates the DisconnectUser WebSocket operation in the Amazon IVS Chat Messaging API.

• send_event — Sends an event to a room. Use this within your application’s business logic to send events to clients of a room; e.g., to notify clients to change the way the chat UI is rendered.

Chat Token Endpoint
• create_chat_token — Creates an encrypted token that is used by a chat participant to establish an individual WebSocket chat connection to a room. When the token is used to connect to chat, the connection is valid for the session duration specified in the request. The token becomes invalid at the token-expiration timestamp included in the response.

Room Endpoints
• create_room — Creates a room that allows clients to connect and pass messages.

• delete_room — Deletes the specified room.

• get_room — Gets the specified room.

• list_rooms — Gets summary information about all your rooms in the AWS region where the API request is processed.

• update_room — Updates a room’s configuration.

Logging Configuration Endpoints
• create_logging_configuration — Creates a logging configuration that allows clients to store and record sent messages.

• delete_logging_configuration — Deletes the specified logging configuration.

• get_logging_configuration — Gets the specified logging configuration.

• list_logging_configurations — Gets summary information about all your logging configurations in the AWS region where the API request is processed.

• update_logging_configuration — Updates a specified logging configuration.

Tags Endpoints
• list_tags_for_resource — Gets information about AWS tags for the specified ARN.
• tag_resource — Adds or updates tags for the AWS resource with the specified ARN.
• untag_resource — Removes tags from the resource with the specified ARN.

All the above are HTTP operations. There is a separate messaging API for managing Chat resources; see the Amazon IVS Chat Messaging API Reference.

Usage

ivschat(config = list(), credentials = list(), endpoint = NULL, region = NULL)

Arguments

config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.
Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- ivschat(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `create_chat_token`: Creates an encrypted token that is used by a chat participant to establish an individual WebSocket chat connection to a room.
- `create_logging_configuration`: Creates a logging configuration that allows clients to store and record sent messages.
- `create_room`: Creates a room that allows clients to connect and pass messages.
- `delete_logging_configuration`: Deletes the specified logging configuration.
- `delete_message`: Sends an event to a specific room which directs clients to delete a specific message; that is, unrender it from view and delete it from the client’s chat history.
- `delete_room`: Deletes the specified room.
- `disconnect_user`: Disconnects all connections using a specified user ID from a room.
- `get_logging_configuration`: Gets the specified logging configuration.
### Examples

```r
## Not run:
svc <- ivschat()
svc$create_chat_token(
  Foo = 123
)

## End(Not run)
```

---

**ivsrealtime**  
*Amazon Interactive Video Service RealTime*

### Description

#### Introduction

The Amazon Interactive Video Service (IVS) real-time API is REST compatible, using a standard HTTP API and an AWS EventBridge event stream for responses. JSON is used for both requests and responses, including errors.

**Terminology:**

- A **stage** is a virtual space where participants can exchange video in real time.
- A **participant token** is a token that authenticates a participant when they join a stage.
- A **participant object** represents participants (people) in the stage and contains information about them. When a token is created, it includes a participant ID; when a participant uses that token to join a stage, the participant is associated with that participant ID. There is a 1:1 mapping between participant tokens and participants.
- Server-side composition: The **composition** process composites participants of a stage into a single video and forwards it to a set of outputs (e.g., IVS channels). Composition endpoints support this process.
- Server-side composition: A **composition** controls the look of the outputs, including how participants are positioned in the video.
Resources

The following resources contain information about your IVS live stream (see Getting Started with Amazon IVS Real-Time Streaming):

- **Stage** — A stage is a virtual space where participants can exchange video in real time.

Tagging

A tag is a metadata label that you assign to an AWS resource. A tag comprises a key and a value, both set by you. For example, you might set a tag as `topic:nature` to label a particular video category. See Tagging AWS Resources for more information, including restrictions that apply to tags and "Tag naming limits and requirements"; Amazon IVS stages has no service-specific constraints beyond what is documented there.

Tags can help you identify and organize your AWS resources. For example, you can use the same tag for different resources to indicate that they are related. You can also use tags to manage access (see Access Tags).

The Amazon IVS real-time API has these tag-related endpoints: `tag_resource`, `untag_resource`, and `list_tags_for_resource`. The following resource supports tagging: Stage.

At most 50 tags can be applied to a resource.

**Stages Endpoints**

- `create_participant_token` — Creates an additional token for a specified stage. This can be done after stage creation or when tokens expire.
- `create_stage` — Creates a new stage (and optionally participant tokens).
- `delete_stage` — Shuts down and deletes the specified stage (disconnecting all participants).
- `disconnect_participant` — Disconnects a specified participant and revokes the participant permanently from a specified stage.
- `get_participant` — Gets information about the specified participant token.
- `get_stage` — Gets information for the specified stage.
- `get_stage_session` — Gets information for the specified stage session.
- `list_participant_events` — Lists events for a specified participant that occurred during a specified stage session.
- `list_participants` — Lists all participants in a specified stage session.
- `list_stages` — Gets summary information about all stages in your account, in the AWS region where the API request is processed.
- `list_stage_sessions` — Gets all sessions for a specified stage.
- `update_stage` — Updates a stage’s configuration.

**Composition Endpoints**

- `get_composition` — Gets information about the specified Composition resource.
- `list_compositions` — Gets summary information about all Compositions in your account, in the AWS region where the API request is processed.
- `start_composition` — Starts a Composition from a stage based on the configuration provided in the request.
• stop_composition — Stops and deletes a Composition resource. Any broadcast from the Composition resource is stopped.

**EncoderConfiguration Endpoints**

• create_encoder_configuration — Creates an EncoderConfiguration object.
• delete_encoder_configuration — Deletes an EncoderConfiguration resource. Ensures that no Compositions are using this template; otherwise, returns an error.
• get_encoder_configuration — Gets information about the specified EncoderConfiguration resource.
• list_encoder_configurations — Gets summary information about all EncoderConfigurations in your account, in the AWS region where the API request is processed.

**StorageConfiguration Endpoints**

• create_storage_configuration — Creates a new storage configuration, used to enable recording to Amazon S3.
• delete_storage_configuration — Deletes the storage configuration for the specified ARN.
• get_storage_configuration — Gets the storage configuration for the specified ARN.
• list_storage_configurations — Gets summary information about all storage configurations in your account, in the AWS region where the API request is processed.

**Tags Endpoints**

• list_tags_for_resource — Gets information about AWS tags for the specified ARN.
• tag_resource — Adds or updates tags for the AWS resource with the specified ARN.
• untag_resource — Removes tags from the resource with the specified ARN.

**Usage**

```python
ivsrealtime(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

**Arguments**

- **config**
  - Optional configuration of credentials, endpoint, and/or region.
  - **credentials:**
    - **creds:**
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
– **anonymous**: Set anonymous credentials.

- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endsWith.html.

**credentials**

Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.

- **anonymous**: Set anonymous credentials.

**endpoint**

Optional shorthand for complete URL to use for the constructed client.

**region**

Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- ivsrealtime(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
  
```
stsRegionalEndpoint = "string"
),
credentials = list(
    creds = list(
        accessKeyId = "string",
        secretAccessKey = "string",
        sessionToken = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

create_encoder_configuration Creates an EncoderConfiguration object
create_participant_token Creates an additional token for a specified stage
create_stage Creates a new stage (and optionally participant tokens)
create_storage_configuration Creates a new storage configuration, used to enable recording to Amazon S3
delete_encoder_configuration Deletes an EncoderConfiguration resource
delete_participant_disconnect_participant Disconnects a specified participant and revokes the participant permanently from a specified stage
delete_storage_configuration Deletes the storage configuration for the specified ARN
disconnect_participant Disconnects a specified participant and revokes the participant permanently from a specified stage
disconnect_participant_events Lists events for a specified participant that occurred during a specified stage session
disconnect_participants Lists all participants in a specified stage session
disconnect_participants_in_a_specified_stage session
get_composition Get information about the specified Composition resource
get_encoder_configuration resource
get_encoder_configuration resource
get_participant Gets information about the specified participant token
get_participant Gets information for the specified stage
get_participant Gets information for the specified stage session
get_participants Gets information for the specified stage
get_storage_configuration for the specified ARN
get_storage_configuration for the specified ARN
get_stage Gets information about the specified Composition resource
get_storage_configuration Gets information about the specified Storage Configuration
get_storage_configuration
get_storage_session Get information about the specified Storage Session
get_storage_session
list_compositions Gets summary information about all Compositions in your account, in the AWS region where the API request is processed
list_encoder_configurations Gets summary information about all EncoderConfigurations in your account, in the AWS region where the API request is processed
list_participant_events Lists events for a specified participant that occurred during a specified stage session
list_participants Lists all participants in a specified stage session
list_stages Gets summary information about all stages in your account, in the AWS region where the API request is processed
list_storage_configurations Gets summary information about all storage configurations in your account, in the AWS region where the API request is processed
list_tags_for_resource Gets information about AWS tags for the specified ARN
list_tags_for_resource
start_composition Starts a Composition from a stage based on the configuration provided in the request
stop_composition Stops and deletes a Composition resource
tag_resource Adds or updates tags for the AWS resource with the specified ARN
untag_resource Removes tags from the resource with the specified ARN
update_stage Updates a stage’s configuration
Examples

```r
## Not run:
svc <- ivsrealtime()
svc$create_encoder_configuration(
  Foo = 123
)
## End(Not run)
```

---

**kafka**

*Managed Streaming for Kafka*

**Description**

The operations for managing an Amazon MSK cluster.

**Usage**

```r
kafka(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

**Arguments**

- `config` Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
    - `creds`:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
    - `anonymous`: Set anonymous credentials.
  - `endpoint`: The complete URL to use for the constructed client.
  - `region`: The AWS Region used in instantiating the client.
  - `close_connection`: Immediately close all HTTP connections.
  - `timeout`: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - `s3_force_path_style`: Set this to true to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.

- `credentials` Optional credentials shorthand for the config parameter
  - `creds`:
- **access_key_id**: AWS access key ID
- **secret_access_key**: AWS secret access key
- **session_token**: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Optional shorthand for complete URL to use for the constructed client.</strong></td>
<td><code>endpoint</code></td>
</tr>
<tr>
<td><strong>Optional shorthand for AWS Region used in instantiating the client.</strong></td>
<td><code>region</code></td>
</tr>
</tbody>
</table>

**Service syntax**

```r
svc <- kafka(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```
### Operations

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>batch_associate_scram_secret</td>
<td>Associates one or more Scram Secrets with an Amazon MSK cluster</td>
</tr>
<tr>
<td>batch_disassociate_scram_secret</td>
<td>Disassociates one or more Scram Secrets from an Amazon MSK cluster</td>
</tr>
<tr>
<td>create_cluster</td>
<td>Creates a new MSK cluster</td>
</tr>
<tr>
<td>create_cluster_v2</td>
<td>Creates a new MSK cluster</td>
</tr>
<tr>
<td>create_configuration</td>
<td>Creates a new MSK configuration</td>
</tr>
<tr>
<td>create_replicator</td>
<td>Creates the replicator</td>
</tr>
<tr>
<td>create_vpc_connection</td>
<td>Creates a new MSK VPC connection</td>
</tr>
<tr>
<td>delete_cluster</td>
<td>Deletes the MSK cluster specified by the Amazon Resource Name (ARN) in the request</td>
</tr>
<tr>
<td>delete_cluster_policy</td>
<td>Deletes the MSK cluster policy specified by the Amazon Resource Name (ARN) in the request</td>
</tr>
<tr>
<td>delete_configuration</td>
<td>Deletes an MSK Configuration</td>
</tr>
<tr>
<td>delete_replicator</td>
<td>Deletes a replicator</td>
</tr>
<tr>
<td>delete_vpc_connection</td>
<td>Deletes a MSK VPC connection</td>
</tr>
<tr>
<td>describe_cluster</td>
<td>Returns a description of the MSK cluster whose Amazon Resource Name (ARN) is specified</td>
</tr>
<tr>
<td>describe_cluster_v2</td>
<td>Returns a description of the cluster operation specified by the ARN</td>
</tr>
<tr>
<td>describe_configuration_v2</td>
<td>Returns a description of the cluster operation specified by the ARN</td>
</tr>
<tr>
<td>describe_replicator</td>
<td>Returns a description of this MSK configuration</td>
</tr>
<tr>
<td>describe_vpc_connection</td>
<td>Returns a description of this MSK VPC connection</td>
</tr>
<tr>
<td>get_bootstrap_brokers</td>
<td>A list of brokers that a client application can use to bootstrap</td>
</tr>
<tr>
<td>get_client_vpc_connections</td>
<td>Get the MSK cluster policy specified by the Amazon Resource Name (ARN) in the request</td>
</tr>
<tr>
<td>get_compatible_kafka_versions</td>
<td>Gets the Apache Kafka versions to which you can update the MSK cluster</td>
</tr>
<tr>
<td>get_cluster_operation_v2</td>
<td>Returns a description of the cluster operation specified by the ARN</td>
</tr>
<tr>
<td>list_client_vpc_connections</td>
<td>Returns a list of all the operations that have been performed on the specified MSK cluster</td>
</tr>
<tr>
<td>list_cluster_operations</td>
<td>Returns a list of all the operations that have been performed on the specified MSK cluster</td>
</tr>
<tr>
<td>list_cluster_operations_v2</td>
<td>Returns a list of all the MSK clusters in the current Region</td>
</tr>
<tr>
<td>list_clusters</td>
<td>Returns a list of all the MSK clusters in the current Region</td>
</tr>
<tr>
<td>list_clusters_v2</td>
<td>Returns a list of all the MSK clusters in the current Region</td>
</tr>
<tr>
<td>list_configurations</td>
<td>Returns a list of all the MSK configurations in this Region</td>
</tr>
<tr>
<td>list_configurations_v2</td>
<td>Returns a list of all the MSK configurations in this Region</td>
</tr>
<tr>
<td>list_kafka_versions</td>
<td>Returns a list of Apache Kafka versions</td>
</tr>
<tr>
<td>list_nodes</td>
<td>Returns a list of the broker nodes in the cluster</td>
</tr>
<tr>
<td>list_replicators</td>
<td>Lists the replicators</td>
</tr>
<tr>
<td>list_scram_secrets</td>
<td>Returns a list of the Scram Secrets associated with an Amazon MSK cluster</td>
</tr>
<tr>
<td>list_tags_for_resource</td>
<td>Returns a list of the tags associated with the specified resource</td>
</tr>
<tr>
<td>list_vpc_connections</td>
<td>Returns a list of all the VPC connections in this Region</td>
</tr>
<tr>
<td>put_cluster_policy</td>
<td>Creates or updates the MSK cluster policy specified by the cluster Amazon Resource Name</td>
</tr>
<tr>
<td>reboot_broker</td>
<td>Reboots brokers</td>
</tr>
<tr>
<td>reject_client_vpc_connection</td>
<td>Returns empty response</td>
</tr>
<tr>
<td>tag_resource</td>
<td>Adds tags to the specified MSK resource</td>
</tr>
<tr>
<td>untag_resource</td>
<td>Removes the tags associated with the keys that are provided in the query</td>
</tr>
<tr>
<td>update_broker_count</td>
<td>Updates the number of broker nodes in the cluster</td>
</tr>
<tr>
<td>update_broker_storage</td>
<td>Updates the EBS storage associated with MSK brokers</td>
</tr>
<tr>
<td>update_broker_type</td>
<td>Updates EC2 instance type</td>
</tr>
<tr>
<td>update_cluster_configuration</td>
<td>Updates the cluster with the configuration that is specified in the request body</td>
</tr>
<tr>
<td>update_cluster_kafka_version</td>
<td>Updates the Apache Kafka version for the cluster</td>
</tr>
</tbody>
</table>
**kafkaconnect**

- **update_configuration**: Updates an MSK configuration
- **update_connectivity**: Updates the cluster’s connectivity configuration
- **update_monitoring**: Updates the monitoring settings for the cluster
- **update_replication_info**: Updates replication info of a replicator
- **update_security**: Updates the security settings for the cluster
- **update_storage**: Updates cluster broker volume size (or) sets cluster storage mode to TIERED

**Examples**

```r
t## Not run:
svc <- kafka()
svc$batch_associate_scram_secret(
   Foo = 123
)

t## End(Not run)
```

---

**kafkaconnect**  
**Managed Streaming for Kafka Connect**

**Description**

Managed Streaming for Kafka Connect

**Usage**

```r
kafkaconnect(
   config = list(),
   credentials = list(),
   endpoint = NULL,
   region = NULL
)
```

**Arguments**

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.
- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials**

Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint**

Optional shorthand for complete URL to use for the constructed client.

**region**

Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- kafkaconnect(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
  
```
sts_regional_endpoint = "string"
),
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

create_connector
create_custom_plugin
create_worker_configuration
delete_connector
delete_custom_plugin
delete_worker_configuration
describe_connector
describe_custom_plugin
describe_worker_configuration
list_connectors
list_custom_plugins
list_tags_for_resource
list_worker_configurations
tag_resource
untag_resource
update_connector

Examples

## Not run:
svc <- kafkaconnect()
svc$create_connector(
    Foo = 123
)

## End(Not run)
Amazon Kendra is a service for indexing large document sets.

Usage

kendra(config = list(), credentials = list(), endpoint = NULL, region = NULL)

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**: Optional credentials shorthand for the config parameter
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
    - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

- **endpoint**: Optional shorthand for complete URL to use for the constructed client.

- **region**: Optional shorthand for AWS Region used in instantiating the client.
Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- kendra(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `associate_entities_to_experience`: Grants users or groups in your IAM Identity Center identity source access to your Amazon Kendra experience.
- `associate_personas_to_entities`: Defines the specific permissions of users or groups in your IAM Identity Center identity source with access to your Amazon Kendra experience.
- `batch_delete_document`: Removes one or more documents from an index.
- `batch_delete_featured_results_set`: Removes one or more sets of featured results.
- `batch_get_document_status`: Returns the indexing status for one or more documents submitted with the BatchPutDocument API.
- `batch_put_document`: Adds one or more documents to an index.
- `clear_query_suggestions`: Clears existing query suggestions from an index.
- `create_access_control_configuration`: Creates an access configuration for your documents.
create_data_source
create_experience
create_faq
create_featured_results_set
create_index
create_query_suggestions_block_list
create_thesaurus
delete_access_control_configuration
delete_data_source
delete_experience
delete_faq
delete_index
delete_principal_mapping
delete_query_suggestions_block_list
delete_thesaurus
describe_access_control_configuration
describe_data_source
describe_experience
describe_faq
describe_featured_results_set
describe_index
describe_principal_mapping
describe_query_suggestions_block_list
describe_query_suggestions_config
describe_thesaurus
disassociate_entities_from_experience
disassociate_personas_from_entities
get_query_suggestions
get_snapshots
list_access_control_configurations
list_data_sources
list_data_source_sync_jobs
list_entity_personas
list_experience_entities
list_experiences
list_faqs
list_featured_results_sets
list_groups_older_than_ordering_id
list_indices
list_query_suggestions_block_lists
list_tags_for_resource
list_thesauri
put_principal_mapping
query
retrieve
start_data_source_sync_job
stop_data_source_sync_job
submit_feedback

create_data_source
create_experience
create_faq
create_featured_results_set
create_index
create_query_suggestions_block_list
create_thesaurus
delete_access_control_configuration
delete_data_source
delete_experience
delete_faq
delete_index
delete_principal_mapping
delete_query_suggestions_block_list
delete_thesaurus
describe_access_control_configuration
describe_data_source
describe_experience
describe_faq
describe_featured_results_set
describe_index
describe_principal_mapping
describe_query_suggestions_block_list
describe_query_suggestions_config
describe_thesaurus
disassociate_entities_from_experience
disassociate_personas_from_entities
get_query_suggestions
get_snapshots
list_access_control_configurations
list_data_sources
list_data_source_sync_jobs
list_entity_personas
list_experience_entities
list_experiences
list_faqs
list_featured_results_sets
list_groups_older_than_ordering_id
list_indices
list_query_suggestions_block_lists
list_tags_for_resource
list_thesauri
put_principal_mapping
query
retrieve
start_data_source_sync_job
stop_data_source_sync_job
submit_feedback

Creates a data source connector that you want to use with an Amazon Kendra index
Creates an Amazon Kendra experience such as a search application
Creates a set of frequently ask questions (FAQs) using a specified FAQ file stored in an Amazon S3 bucket
Creates a set of featured results to display at the top of the search results page
Creates an Amazon Kendra index
Creates a block list to exclude certain queries from suggestions
Creates a thesaurus for an index
Deletes an access control configuration that you created for your documents in an index
Deletes an Amazon Kendra data source connector
Deletes your Amazon Kendra experience such as a search application
Removes an FAQ from an index
Deletes an Amazon Kendra index
Deletes a group so that all users and sub groups that belong to the group can no longer access documents only available to that group
Deletes a block list used for query suggestions for an index
Deletes an Amazon Kendra thesaurus
Gets information about an access control configuration that you created for your documents in an index
Gets information about your Amazon Kendra experience such as a search application
Gets information about an FAQ list
Gets information about a set of featured results
Gets information about an Amazon Kendra index
Describes the processing of PUT and DELETE actions for mapping users to their groups
Gets information about a block list used for query suggestions for an index
Gets information on the settings of query suggestions for an index
Gets information about an Amazon Kendra thesaurus
Prevents users or groups in your IAM Identity Center identity source from accessing your Amazon Kendra experience
Removes the specific permissions of users or groups in your IAM Identity Center identity source
Fetches the queries that are suggested to your users
Retrieves search metrics data
Lists one or more access control configurations for an index
Lists the data source connectors that you have created
Gets statistics about synchronizing a data source connector
Lists specific permissions of users and groups with access to your Amazon Kendra experience
Lists users or groups in your IAM Identity Center identity source that are granted access to your Amazon Kendra experiences
Gets a list of FAQ lists associated with an index
Lists all your sets of featured results for a given index
Provides a list of groups that are mapped to users before a given ordering or timestamp
Lists the Amazon Kendra indexes that you created
Lists the block lists used for query suggestions for an index
Gets a list of tags associated with a specified resource
Lists the thesauri for an index
Maps users to their groups so that you only need to provide the user ID when you issue the query
Searches an index given an input query
Retrieves relevant passages or text excerpts given an input query
Starts a synchronization job for a data source connector
Stops a synchronization job that is currently running
Enables you to provide feedback to Amazon Kendra to improve the performance of
tag_resource
untag_resource
update_access_control_configuration
update_data_source
update_experience
update_featured_results_set
update_index
update_query_suggestions_block_list
update_query_suggestions_config
update_thesaurus

kendraranking

Amazon Kendra Intelligent Ranking

Description
Amazon Kendra Intelligent Ranking uses Amazon Kendra semantic search capabilities to intelligently re-rank a search service’s results.

Usage
kendraranking(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)

Arguments
config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID

Examples
## Not run:
svc <- kendra()
svc$associate_entities_to_experience(
  Foo = 123
)
## End(Not run)
credentials

Optional credentials shorthand for the config parameter

- creds:
  - access_key_id: AWS access key ID
  - secret_access_key: AWS secret access key
  - session_token: AWS temporary session token
- profile: The name of a profile to use. If not given, then the default profile is used.
- anonymous: Set anonymous credentials.

endpoint

Optional shorthand for complete URL to use for the constructed client.

region

Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- kendraranking(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string"
  ),
```
keyspaces

region = "string",
close_connection = "logical",
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

create_rescore_execution_plan                  Creates a rescore execution plan
delete_rescore_execution_plan                 Deletes a rescore execution plan
describe_rescore_execution_plan               Gets information about a rescore execution plan
list_rescore_execution_plans                 Lists your rescore execution plans
list_tags_for_resource                         Gets a list of tags associated with a specified resource
rescore                                        Rescores or re-ranks search results from a search service such as OpenSearch (self managed)
tag_resource                                    Adds a specified tag to a specified rescore execution plan
untag_resource                                  Removes a tag from a rescore execution plan
update_rescore_execution_plan                 Updates a rescore execution plan

Examples

## Not run:
svc <- kendraranking()
svc$create_rescore_execution_plan(
    Foo = 123
)

## End(Not run)
Description

Amazon Keyspaces (for Apache Cassandra) is a scalable, highly available, and managed Apache Cassandra-compatible database service. Amazon Keyspaces makes it easy to migrate, run, and scale Cassandra workloads in the Amazon Web Services Cloud. With just a few clicks on the Amazon Web Services Management Console or a few lines of code, you can create keyspaces and tables in Amazon Keyspaces, without deploying any infrastructure or installing software.

In addition to supporting Cassandra Query Language (CQL) requests via open-source Cassandra drivers, Amazon Keyspaces supports data definition language (DDL) operations to manage keyspaces and tables using the Amazon Web Services SDK and CLI, as well as infrastructure as code (IaC) services and tools such as CloudFormation and Terraform. This API reference describes the supported DDL operations in detail.

For the list of all supported CQL APIs, see Supported Cassandra APIs, operations, and data types in Amazon Keyspaces in the Amazon Keyspaces Developer Guide.

To learn how Amazon Keyspaces API actions are recorded with CloudTrail, see Amazon Keyspaces information in CloudTrail in the Amazon Keyspaces Developer Guide.

For more information about Amazon Web Services APIs, for example how to implement retry logic or how to sign Amazon Web Services API requests, see Amazon Web Services APIs in the General Reference.

Usage

```python
keyspaces(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

Arguments

- `config`: Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
    - `creds`:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
    - `anonymous`: Set anonymous credentials.
  - `endpoint`: The complete URL to use for the constructed client.
  - `region`: The AWS Region used in instantiating the client.
  - `close_connection`: Immediately close all HTTP connections.
  - `timeout`: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials**

Optional credentials shorthand for the config parameter

• **creds**:  
  – **access_key_id**: AWS access key ID  
  – **secret_access_key**: AWS secret access key  
  – **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

**endpoint**

Optional shorthand for complete URL to use for the constructed client.

**region**

Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- keyspaces(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    )
  
```
### Operations

- **create_keyspace**
  - The CreateKeyspace operation adds a new keyspace to your account.

- **create_table**
  - The CreateTable operation adds a new table to the specified keyspace.

- **delete_keyspace**
  - The DeleteKeyspace operation deletes a keyspace and all of its tables.

- **delete_table**
  - The DeleteTable operation deletes a table and all of its data.

- **get_keyspace**
  - Returns the name and the Amazon Resource Name (ARN) of the specified table.

- **get_table**
  - Returns information about the table, including the table’s name and current status, the keyspace name, configuration settings, and metadata.

- **get_table_auto_scaling_settings**
  - Returns auto scaling related settings of the specified table in JSON format.

- **list_keyspaces**
  - Returns a list of keyspaces.

- **list_tables**
  - Returns a list of tables for a specified keyspace.

- **list_tags_for_resource**
  - Returns a list of all tags associated with the specified Amazon Keyspaces resource.

- **restore_table**
  - Restores the table to the specified point in time within the earliest_restorable_timestamp and the current time.

- **tag_resource**
  - Associates a set of tags with a Amazon Keyspaces resource.

- **untag_resource**
  - Removes the association of tags from a Amazon Keyspaces resource.

- **update_table**
  - Adds new columns to the table or updates one of the table’s settings, for example capacity mode, auto scaling, encryption, point-in-time recovery, or ttl settings.

### Examples

```r
## Not run:
svc <- keyspaces()
svc$create_keyspace(
  Foo = 123
)
## End(Not run)
```

---

### Description

Amazon Kinesis

Amazon Kinesis Data Streams is a managed service that scales elastically for real-time processing of streaming big data.
Usage

kinesis(config = list(), credentials = list(), endpoint = NULL, region = NULL)

Arguments

config Optional configuration of credentials, endpoint, and/or region.

• credentials:
  – creds:
    * access_key_id: AWS access key ID
    * secret_access_key: AWS secret access key
    * session_token: AWS temporary session token
  – profile: The name of a profile to use. If not given, then the default profile is used.
  – anonymous: Set anonymous credentials.

• endpoint: The complete URL to use for the constructed client.

• region: The AWS Region used in instantiating the client.

• close_connection: Immediately close all HTTP connections.

• timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

• s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

• sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter

• creds:
  – access_key_id: AWS access key ID
  – secret_access_key: AWS secret access key
  – session_token: AWS temporary session token

• profile: The name of a profile to use. If not given, then the default profile is used.

• anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```r
csvc <- kinesis(
    config = list(
        credentials = list(
            creds = list(
                access_key_id = "string",
                secret_access_key = "string",
                session_token = "string"
            ),
            profile = "string",
            anonymous = "logical"
        ),
        endpoint = "string",
        region = "string",
        close_connection = "logical",
        timeout = "numeric",
        s3_force_path_style = "logical",
        sts_regional_endpoint = "string"
    ),
    credentials = list(
        creds = list(
            access_key_id = "string",
            secret_access_key = "string",
            session_token = "string"
        ),
        profile = "string",
        anonymous = "logical"
    ),
    endpoint = "string",
    region = "string"
)
```

Operations

- **add_tags_to_stream**: Adds or updates tags for the specified Kinesis data stream
- **create_stream**: Creates a Kinesis data stream
- **decrease_stream_retention_period**: Decreases the Kinesis data stream’s retention period, which is the length of time data records are accessible after they are added to the stream
- **delete_resource_policy**: Deletes a policy for the specified data stream or consumer
- **delete_stream**: Deletes a Kinesis data stream and all its shards and data
- **deregister_stream_consumer**: To deregister a consumer, provide its ARN
- **describe_limits**: Describes the shard limits and usage for the account
- **describe_stream**: Describes the specified Kinesis data stream
- **describe_stream_consumer**: To get the description of a registered consumer, provide the ARN of the consumer
- **describe_stream_summary**: Provides a summarized description of the specified Kinesis data stream without the shard list
- **disable_enhanced_monitoring**: Disables enhanced monitoring
- **enable_enhanced_monitoring**: Enables enhanced Kinesis data stream monitoring for shard-level metrics
- **get_records**: Gets data records from a Kinesis data stream’s shard
- **get_resource_policy**: Returns a policy attached to the specified data stream or consumer
get_shard_iterator
increase_stream_retention_period
list_shards
list_stream_consumers
list_streams
list_tags_for_stream
merge_shards
put_record
put_records
put_resource_policy
register_stream_consumer
remove_tags_from_stream
split_shard
start_stream_encryption
stop_stream_encryption
update_shard_count
update_stream_mode

Gets an Amazon Kinesis shard iterator
Increases the Kinesis data stream’s retention period, which is the length of time data records are accessible after they are added to the stream
Lists the shards in a stream and provides information about each shard
Lists the consumers registered to receive data from a stream using enhanced fan-out, and
Lists your Kinesis data streams
Lists the tags for the specified Kinesis data stream
Merges two adjacent shards in a Kinesis data stream and combines them into a single shard
Writes a single data record into an Amazon Kinesis data stream
Writes multiple data records into a Kinesis data stream in a single call (also referred to as a PutRecords request)
Attaches a resource-based policy to a data stream or registered consumer
Registers a consumer with a Kinesis data stream
Removes tags from the specified Kinesis data stream
Splits a shard into two new shards in the Kinesis data stream, to increase the stream’s capacity
Enables or updates server-side encryption using an Amazon Web Services KMS key for a specified stream
Disables server-side encryption for a specified stream
Updates the shard count of the specified stream to the specified number of shards
Updates the capacity mode of the data stream

Examples

```r
## Not run:
svc <- kinesis()
svc$add_tags_to_stream(
  Foo = 123
)
## End(Not run)
```

---

# kinesisanalytics

---

## Description

### Overview

This documentation is for version 1 of the Amazon Kinesis Data Analytics API, which only supports SQL applications. Version 2 of the API supports SQL and Java applications. For more information about version 2, see Amazon Kinesis Data Analytics API V2 Documentation.

This is the *Amazon Kinesis Analytics v1 API Reference*. The Amazon Kinesis Analytics Developer Guide provides additional information.
Usage

```r
kinesisanalytics(  
    config = list(),  
    credentials = list(),  
    endpoint = NULL,  
    region = NULL
)
```

Arguments

- `config` Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - `creds`:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
    - `anonymous`: Set anonymous credentials.
  - `endpoint`: The complete URL to use for the constructed client.
  - `region`: The AWS Region used in instantiating the client.
  - `close_connection`: Immediately close all HTTP connections.
  - `timeout`: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - `s3_force_path_style`: Set this to `true` to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  - `sts_regional_endpoint`: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-_train.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-_train.html)

- `credentials` Optional credentials shorthand for the `config` parameter
  - `creds`:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
    - `anonymous`: Set anonymous credentials.

- `endpoint` Optional shorthand for complete URL to use for the constructed client.

- `region` Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```r
svc <- kinesisanalytics(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `add_application_cloud_watch_logging_option`
- `add_application_input`
- `add_application_input_processing_configuration`
- `add_application_output`
- `add_application_reference_data_source`
- `create_application`
- `delete_application`
- `delete_application_cloud_watch_logging_option`
- `delete_application_input_processing_configuration`
- `delete_application_output`
- `delete_application_reference_data_source`
- `describe_application`
- `discover_input_schema`
- `list_applications`

This documentation is for version 1 of the Amazon Kinesis Data Analytics.
list_tags_for_resource
start_application
stop_application
tag_resource
untag_resource
update_application

Examples

```r
## Not run:
svc <- kinesisanalytics()
svc$add_application_cloud_watch_logging_option(
  Foo = 123
)

## End(Not run)
```

Amazon Kinesis Analytics

Description

Amazon Managed Service for Apache Flink was previously known as Amazon Kinesis Data Analytics for Apache Flink.

Amazon Managed Service for Apache Flink is a fully managed service that you can use to process and analyze streaming data using Java, Python, SQL, or Scala. The service enables you to quickly author and run Java, SQL, or Scala code against streaming sources to perform time series analytics, feed real-time dashboards, and create real-time metrics.

Usage

```r
kinesisanalyticsv2(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
• **secret_access_key**: AWS secret access key
• **session_token**: AWS temporary session token
  – **profile**: The name of a profile to use. If not given, then the default profile is used.
  – **anonymous**: Set anonymous credentials.

• **endpoint**: The complete URL to use for the constructed client.
• **region**: The AWS Region used in instantiating the client.
• **close_connection**: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-]](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized.html)

**credentials** Optional credentials shorthand for the config parameter

• **creds**:
  – **access_key_id**: AWS access key ID
  – **secret_access_key**: AWS secret access key
  – **session_token**: AWS temporary session token
  – **profile**: The name of a profile to use. If not given, then the default profile is used.
  – **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- kinesisanalyticsv2(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
```
Region = "string",
close_connection = "logical",
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
)
,
credentials = list(
  creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
  ),
  profile = "string",
  anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

add_application_cloud_watch_logging_option
add_application_input
add_application_input_processing_configuration
add_application_output
add_application_reference_data_source
add_application_vpc_configuration
create_application
create_application_presigned_url
create_application_snapshot
delete_application
delete_application_cloud_watch_logging_option
delete_application_input_processing_configuration
delete_application_output
delete_application_reference_data_source
delete_application_snapshot
delete_application_vpc_configuration
describe_application
describe_application_snapshot
describe_application_version
discover_input_schema
list_applications
list_application_snapshots
list_application_versions
list_tags_for_resource
rollback_application
start_application
stop_application

Adds an Amazon CloudWatch log stream to monitor application configuration changes.
Adds a streaming source to your SQL-based Kinesis Data Analytics application.
Adds an InputProcessingConfiguration to a SQL-based Kinesis Data Analytics application.
Adds an external destination to your SQL-based Kinesis Data Analytics application.
Adds a reference data source to an existing SQL-based Kinesis Data Analytics application.
Creates a Virtual Private Cloud (VPC) configuration to the application.
Creates and returns a URL that you can use to connect to an application’s extension.
Creates a snapshot of the application’s state data.
Deletes the specified application.
Deletes an Amazon CloudWatch log stream from an SQL-based Kinesis Data Analytics application.
Deletes an InputProcessingConfiguration from an input.
Deletes the output destination configuration from your SQL-based Kinesis Data Analytics application.
Deletes a reference data source configuration from the specified SQL-based Kinesis Data Analytics application.
Deletes a snapshot of application state.
Removes a VPC configuration from a Managed Service for Apache Flink application.
Returns information about a specific Managed Service for Apache Flink application.
Returns information about a snapshot of application state data.
Provides a detailed description of a specified version of the application.
Infers a schema for a SQL-based Kinesis Data Analytics application by examining sample records.
Returns a list of Managed Service for Apache Flink applications in your account.
Lists information about the current application snapshots.
Lists all the versions for the specified application, including versions that were rolled back.
Retrieves the list of key-value tags assigned to the application.
Reverts the application to the previous running version.
Starts the specified Managed Service for Apache Flink application.
Stops the application from processing data.
tag_resource
untag_resource
update_application
update_application_maintenance_configuration

Add one or more key-value tags to a Managed Service for Apache Flink application.
Removes one or more tags from a Managed Service for Apache Flink application.
Updates an existing Managed Service for Apache Flink application.
Updates the maintenance configuration of the Managed Service for Apache Flink application.

Examples

```r
## Not run:
svc <- kinesisanalyticsv2()
svc$add_application_cloud_watch_logging_option(
  Foo = 123
)

## End(Not run)
```

**Description**

Key Management Service

Key Management Service (KMS) is an encryption and key management web service. This guide describes the KMS operations that you can call programmatically. For general information about KMS, see the [Key Management Service Developer Guide](#).

KMS has replaced the term *customer master key (CMK)* with *KMS key* and *KMS key*. The concept has not changed. To prevent breaking changes, KMS is keeping some variations of this term.

Amazon Web Services provides SDKs that consist of libraries and sample code for various programming languages and platforms (Java, Ruby, .Net, macOS, Android, etc.). The SDKs provide a convenient way to create programmatic access to KMS and other Amazon Web Services services. For example, the SDKs take care of tasks such as signing requests (see below), managing errors, and retrying requests automatically. For more information about the Amazon Web Services SDKs, including how to download and install them, see [Tools for Amazon Web Services](#).

We recommend that you use the Amazon Web Services SDKs to make programmatic API calls to KMS.

If you need to use FIPS 140-2 validated cryptographic modules when communicating with Amazon Web Services, use the FIPS endpoint in your preferred Amazon Web Services Region. For more information about the available FIPS endpoints, see [Service endpoints](#) in the Key Management Service topic of the [Amazon Web Services General Reference](#).

All KMS API calls must be signed and be transmitted using Transport Layer Security (TLS). KMS recommends you always use the latest supported TLS version. Clients must also support cipher suites with Perfect Forward Secrecy (PFS) such as Ephemeral Diffie-Hellman (DHE) or Elliptic Curve Ephemeral Diffie-Hellman (ECDHE). Most modern systems such as Java 7 and later support these modes.
**Signing Requests**

Requests must be signed using an access key ID and a secret access key. We strongly recommend that you do not use your Amazon Web Services account root access key ID and secret access key for everyday work. You can use the access key ID and secret access key for an IAM user or you can use the Security Token Service (STS) to generate temporary security credentials and use those to sign requests.

All KMS requests must be signed with **Signature Version 4**.

**Logging API Requests**

KMS supports CloudTrail, a service that logs Amazon Web Services API calls and related events for your Amazon Web Services account and delivers them to an Amazon S3 bucket that you specify. By using the information collected by CloudTrail, you can determine what requests were made to KMS, who made the request, when it was made, and so on. To learn more about CloudTrail, including how to turn it on and find your log files, see the CloudTrail User Guide.

**Additional Resources**

For more information about credentials and request signing, see the following:

- **Amazon Web Services Security Credentials** - This topic provides general information about the types of credentials used to access Amazon Web Services.
- **Temporary Security Credentials** - This section of the IAM User Guide describes how to create and use temporary security credentials.
- **Signature Version 4 Signing Process** - This set of topics walks you through the process of signing a request using an access key ID and a secret access key.

**Commonly Used API Operations**

Of the API operations discussed in this guide, the following will prove the most useful for most applications. You will likely perform operations other than these, such as creating keys and assigning policies, by using the console.

- encrypt
- decrypt
- generate_data_key
- generate_data_key_without_plaintext

**Usage**

```markdown
kms(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

**Arguments**

- `config` Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
    - `creds`:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
– **profile**: The name of a profile to use. If not given, then the default profile is used.
– **anonymous**: Set anonymous credentials.

• **endpoint**: The complete URL to use for the constructed client.
• **region**: The AWS Region used in instantiating the client.
• **close_connection**: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e., `http://s3.amazonaws.com/BUCKET/KEY`.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

**credentials**
Optional credentials shorthand for the config parameter

• **creds**:
  – **access_key_id**: AWS access key ID
  – **secret_access_key**: AWS secret access key
  – **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```
svc <- kms(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
```
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
  creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
  ),
  profile = "string",
  anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

cancel_key_deletion Cancels the deletion of a KMS key
connect_custom_key_store Connects or reconnects a custom key store to its backing key store
create_alias Creates a friendly name for a KMS key
create_custom_key_store Creates a custom key store backed by a key store that you own and manage
create_grant Adds a grant to a KMS key
create_key Creates a unique customer managed KMS key in your Amazon Web Services account and region
decrypt Decrypts ciphertext that was encrypted by a KMS key using any of the following operations:
delete_alias Deletes the specified alias
delete_custom_key_store Deletes a custom key store
delete_imported_key_material Deletes key material that was previously imported
describe_custom_key_stores Gets information about custom key stores in the account and region
describe_key Provides detailed information about a KMS key
disable_key Sets the state of a KMS key to disabled
disable_key_rotation Disables automatic rotation of the key material of the specified symmetric encryption KMS key
disconnect_custom_key_store Disconnects the custom key store from its backing key store
disable_key_rotation
enable_key Sets the key state of a KMS key to enabled
enable_key_rotation Enables automatic rotation of the key material of the specified symmetric encryption KMS key
encrypt Encrypts plaintext of up to 4,096 bytes using a KMS key
generate_data_key Returns a unique symmetric data key for use outside of KMS
generate_data_key_pair Returns a unique asymmetric data key pair for use outside of KMS
generate_data_key_pair_without_plaintext Returns a unique asymmetric data key pair for use outside of KMS
generate_data_key_without_plaintext Returns a unique symmetric data key for use outside of KMS
generate_mac Generates a hash-based message authentication code (HMAC) for a message using an HMAC KMS key
generate_random Returns a random byte string that is cryptographically secure
get_key_policy Gets a key policy attached to the specified KMS key
get_key_rotation_status Provides detailed information about the rotation status for a KMS key, including information about key rotation
get_parameters_for_import Returns the public key and an import token you need to import or reimport key material
get_public_key Returns the public key of an asymmetric KMS key
import_key_material Imports or reimports key material into an existing KMS key that was created with
list_aliases
list_grants
list_key_policies
list_key_rotations
list_keys
list_resource_tags
list_retirable_grants
put_key_policy
re_encrypt
replicate_key
revoke_grant
rotate_key_on_demand
schedule_key_deletion
sign
tag_resource
untag_resource
update_alias
update_custom_key_store
update_key_description
update_primary_region
verify
verify_mac

Gets a list of aliases in the caller’s Amazon Web Services account and region
Gets a list of all grants for the specified KMS key
Gets the names of the key policies that are attached to a KMS key
Returns information about all completed key material rotations for the specified KMS key
Gets a list of all KMS keys in the caller’s Amazon Web Services account and Region
Returns all tags on the specified KMS key
Returns information about all grants in the Amazon Web Services account and Region
Attaches a key policy to the specified KMS key
Decrypts ciphertext and then reencrypts it entirely within KMS
Replicates a multi-Region key into the specified Region
Deletes a grant
Immediately initiates rotation of the key material of the specified symmetric encryption KMS key
Schedules the deletion of a KMS key
Creates a digital signature for a message or message digest by using the private key
Adds or edits tags on a customer managed key
Deletes tags from a customer managed key
Associates an existing KMS alias with a different KMS key
Changes the properties of a custom key store
Updates the description of a KMS key
Changes the primary key of a multi-Region key
Verifies a digital signature that was generated by the Sign operation
Verifies the hash-based message authentication code (HMAC) for a specified message

Examples

```r
## Not run:
svc <- kms()
# The following example cancels deletion of the specified KMS key.
svc$cancel_key_deletion(
  KeyId = "1234abcd-12ab-34cd-56ef-1234567890ab"
)
## End(Not run)
```

lakeformation  
AWS Lake Formation

Description

Lake Formation  
Defines the public endpoint for the Lake Formation service.
lakeformation

Usage

`lakeformation(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)`

Arguments

`config` Optional configuration of credentials, endpoint, and/or region.

- `credentials`:
  - `creds`:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - `profile`: The name of a profile to use. If not given, then the default profile is used.
  - `anonymous`: Set anonymous credentials.

- `endpoint`: The complete URL to use for the constructed client.

- `region`: The AWS Region used in instantiating the client.

- `close_connection`: Immediately close all HTTP connections.

- `timeout`: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

- `s3_force_path_style`: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.

- `sts_regional_endpoint`: Set sts regional endpoint resolver to regional or legacy `https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html`

`credentials` Optional credentials shorthand for the config parameter

- `creds`:
  - `access_key_id`: AWS access key ID
  - `secret_access_key`: AWS secret access key
  - `session_token`: AWS temporary session token

- `profile`: The name of a profile to use. If not given, then the default profile is used.

- `anonymous`: Set anonymous credentials.

`endpoint` Optional shorthand for complete URL to use for the constructed client.

`region` Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```r
csvc <- lakeformation(
  config = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string",
  close_connection = "logical",
  timeout = "numeric",
  s3_force_path_style = "logical",
  sts_regional_endpoint = "string"
 ),
 credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `add_lf_tags_to_resource` Attaches one or more LF-tags to an existing resource
- `assume_decorated_role_with_saml` Allows a caller to assume an IAM role decorated as the SAML user specified in the SAML assertion included in the request
- `batch_grant_permissions` Batch operation to grant permissions to the principal
- `batch_revoke_permissions` Batch operation to revoke permissions from the principal
- `cancel_transaction` Attempts to cancel the specified transaction
- `commit_transaction` Attempts to commit the specified transaction
- `create_data_cells_filter` Creates a data cell filter to allow one to grant access to certain columns on certain rows
- `create_lake_formation_identity_center_configuration` Creates an IAM Identity Center connection with Lake Formation to allow IAM Identity Center users and groups to access Data Catalog resources
- `create_lake_formation_opt_in` Enforce Lake Formation permissions for the given databases, tables, and principals
- `create_lf_tag` Creates an LF-tag with the specified name and values
- `delete_data_cells_filter` Deletes a data cell filter
- `delete_lake_formation_identity_center_configuration` Deletes an IAM Identity Center connection with Lake Formation
- `delete_lake_formation_opt_in` Remove the Lake Formation permissions enforcement of the given databases, tables, and principals
- `delete_lf_tag` Deletes the specified LF-tag given a key name
delete_objects_on_cancel
deregister_resource
describe_lake_formation_identity_center_configuration
describe_resource
describe_transaction
extend_transaction
get_data_cells_filter
get_data_lake_settings
get_effective_permissions_for_path
get_lf_tag
get_query_state
get_query_statistics
get_resource_lf_tags
get_table_objects
get_temporary_glue_partition_credentials
get_temporary_glue_table_credentials
get_work_unit_results
get_work_units
grant_permissions
list_data_cells_filter
list_lake_formation_opt_ins
list_lf_tags
list_permissions
list_resources
list_table_storage_optimizers
list_transactions
put_data_lake_settings
register_resource
remove_lf_tags_from_resource
revoke_permissions
search_databases_by_lf_tags
search_tables_by_lf_tags
start_query_planning
start_transaction
update_data_cells_filter
update_lake_formation_identity_center_configuration
update_lf_tag
update_resource
update_table_objects
update_table_storage_optimizer

For a specific governed table, provides a list of Amazon S3 objects that will be written during the current transaction and that can be automatically deleted if the transaction is canceled

Deregisters the resource as managed by the Data Catalog

Retrieves the instance ARN and application ARN for the connection

Retrieves the current data access role for the given resource registered in Lake Formation

Returns the details of a single transaction

Indicates to the service that the specified transaction is still active and should not be treated as idle and aborted

Returns a data cells filter

Retrieves the list of the data lake administrators of a Lake Formation-managed data lake

Returns the Lake Formation permissions for a specified table or database

Returns an LF-tag definition

Returns the state of a query previously submitted

Retrieves statistics on the planning and execution of a query

Returns the LF-tags applied to a resource

Returns the set of Amazon S3 objects that make up the specified governed table

This API is identical to GetTemporaryTableCredentials except that this is used when the target Data Catalog resource is of type Partition

Allows a caller in a secure environment to assume a role with permission to access Amazon S3

Returns the work units resulting from the query

Retrieves the work units generated by the StartQueryPlanning operation

Grants permissions to the principal to access metadata in the Data Catalog and data organized in underlying data storage such as Amazon S3

Lists all the data cell filters on a table

Retrieve the current list of resources and principals that are opt in to enforce Lake Formation permissions

Lists LF-tags that the requester has permission to view

Returns a list of the principal permissions on the resource, filtered by the permissions of the caller

Lists the resources registered to be managed by the Data Catalog

Returns the configuration of all storage optimizers associated with a table

Returns metadata about transactions and their status

Sets the list of data lake administrators who have admin privileges on the resource

Registers the resource as managed by the Data Catalog

Removes an LF-tag from the resource

Revoke permissions to the principal to access metadata in the Data Catalog

This operation allows a search on DATABASE resources by TagCondition

This operation allows a search on TABLE resources by LFTags

Submits a request to process a query statement

Starts a new transaction and returns its transaction ID

Updates a data cell filter

Updates the IAM Identity Center connection parameters

Updates the list of possible values for the specified LF-tag key

Updates the data access role used for vending access to the given (registered) resource in Lake Formation

Updates the manifest of Amazon S3 objects that make up the specified governed table

Updates the configuration of the storage optimizers for a table

Examples

```
## Not run:
svc <- lakeformation()
svc$add_lf_tags_to_resource(
  Foo = 123
)```
Lambda is a compute service that lets you run code without provisioning or managing servers. Lambda runs your code on a high-availability compute infrastructure and performs all of the administration of the compute resources, including server and operating system maintenance, capacity provisioning and automatic scaling, code monitoring and logging. With Lambda, you can run code for virtually any type of application or backend service. For more information about the Lambda service, see What is Lambda in the Lambda Developer Guide.

The Lambda API Reference provides information about each of the API methods, including details about the parameters in each API request and response.

You can use Software Development Kits (SDKs), Integrated Development Environment (IDE) Toolkits, and command line tools to access the API. For installation instructions, see Tools for Amazon Web Services.

For a list of Region-specific endpoints that Lambda supports, see Lambda endpoints and quotas in the Amazon Web Services General Reference.

When making the API calls, you will need to authenticate your request by providing a signature. Lambda supports signature version 4. For more information, see Signature Version 4 signing process in the Amazon Web Services General Reference.

CA certificates

Because Amazon Web Services SDKs use the CA certificates from your computer, changes to the certificates on the Amazon Web Services servers can cause connection failures when you attempt to use an SDK. You can prevent these failures by keeping your computer’s CA certificates and operating system up-to-date. If you encounter this issue in a corporate environment and do not manage your own computer, you might need to ask an administrator to assist with the update process. The following list shows minimum operating system and Java versions:

- Microsoft Windows versions that have updates from January 2005 or later installed contain at least one of the required CAs in their trust list.
- Mac OS X 10.4 with Java for Mac OS X 10.4 Release 5 (February 2007), Mac OS X 10.5 (October 2007), and later versions contain at least one of the required CAs in their trust list.
- Red Hat Enterprise Linux 5 (March 2007), 6, and 7 and CentOS 5, 6, and 7 all contain at least one of the required CAs in their default trusted CA list.
- Java 1.4.2_12 (May 2006), 5 Update 2 (March 2005), and all later versions, including Java 6 (December 2006), 7, and 8, contain at least one of the required CAs in their default trusted CA list.
When accessing the Lambda management console or Lambda API endpoints, whether through browsers or programmatically, you will need to ensure your client machines support any of the following CAs:

- Amazon Root CA 1
- Starfield Services Root Certificate Authority - G2
- Starfield Class 2 Certification Authority

Root certificates from the first two authorities are available from Amazon trust services, but keeping your computer up-to-date is the more straightforward solution. To learn more about ACM-provided certificates, see Amazon Web Services Certificate Manager FAQs.

Usage

```r
lambda(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

- `config` Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
    - `creds`:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
    - `anonymous`: Set anonymous credentials.
  - `endpoint`: The complete URL to use for the constructed client.
  - `region`: The AWS Region used in instantiating the client.
  - `close_connection`: Immediately close all HTTP connections.
  - `timeout`: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - `s3_force_path_style`: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
  - `sts_regional_endpoint`: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

- `credentials` Optional credentials shorthand for the config parameter
  - `creds`:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - `profile`: The name of a profile to use. If not given, then the default profile is used.
  - `anonymous`: Set anonymous credentials.

- `endpoint` Optional shorthand for complete URL to use for the constructed client.

- `region` Optional shorthand for AWS Region used in instantiating the client.
Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```
svc <- lambda(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `add_layer_version_permission`: Adds permissions to the resource-based policy of a version of an Lambda layer
- `add_permission`: Grants an Amazon Web Service, Amazon Web Services account, or Amazon Web Services organization permission to use a function
- `create_alias`: Creates an alias for a Lambda function version
- `create_code_signing_config`: Creates a code signing configuration
- `create_event_source_mapping`: Creates a mapping between an event source and an Lambda function
- `create_function`: Creates a Lambda function
- `create_function_url_config`: Creates a Lambda function URL with the specified configuration parameters
- `delete_alias`: Deletes a Lambda function alias
lambda

delete_code_signing_config
delete_event_source_mapping
delete_function
delete_function_code_signing_config
delete_function_concurrency
delete_function_event_invoke_config
delete_function_url_config
delete_layer_version
delete_provisioned_concurrency_config
get_account_settings
get_alias
get_code_signing_config
get_event_source_mapping
get_function
get_function_code_signing_config
get_function_concurrency
get_function_configuration
get_function_event_invoke_config
get_function_url_config
get_layer_version
get_layer_version_by_arn
get_layer_version_policy
get_policy
get_provisioned_concurrency_config
get_runtime_management_config
invoke
invoke_async
invoke_with_response_stream
list_aliases
list_code_signing_configs
list_event_source_mappings
list_function_eventInvoke_configs
list_functions
list_functions_by_code_signing_config
list_function_url_configs
list_layers
list_layer_versions
list_provisioned_concurrency_configs
list_tags
list_versions_by_function
publish_layer_version
publish_version
put_function_code_signing_config
put_function_concurrency
put_function_event_invoke_config
put_provisioned_concurrency_config
put_runtime_management_config
remove_layer_version_permission

Deletes the code signing configuration
Deletes an event source mapping
Deletes a Lambda function
Removes the code signing configuration from the function
Removes a concurrent execution limit from a function
Deletes the configuration for asynchronous invocation for a function, version, or alias
Deletes a Lambda function URL
Deletes a version of an Lambda layer
Deletes the provisioned concurrency configuration for a function
Retrieves details about your account’s limits and usage in an Amazon Web Services Region
Returns details about a Lambda function alias
Returns information about the specified code signing configuration
Returns details about an event source mapping
Returns information about the function or function version, with a link to download the deployment package
Returns the code signing configuration for the specified function
Returns details about the reserved concurrency configuration for a function
Returns the version-specific settings of a Lambda function or version
Retrieves the configuration for asynchronous invocation for a function, version, or alias
Returns details about a Lambda function URL
Returns information about a version of an Lambda layer, with a link to download the layer archive
Returns information about a version of an Lambda layer, with a link to download the layer archive
Returns the permission policy for a version of an Lambda layer
Returns the resource-based IAM policy for a function, version, or alias
Retrieves the provisioned concurrency configuration for a function’s alias or version
Retrieves the runtime management configuration for a function’s version
Invokes a Lambda function
For asynchronous function invocation, use Invoke
Configure your Lambda functions to stream response payloads back to clients
Returns a list of aliases for a Lambda function
Returns a list of code signing configurations
Lists event source mappings
Retrieves a list of configurations for asynchronous invocation for a function
Returns a list of Lambda functions, with the version-specific configuration of each
List the functions that use the specified code signing configuration
Returns a list of Lambda function URLs for the specified function
Lists Lambda layers and shows information about the latest version of each
Lists the versions of an Lambda layer
Retrieves a list of provisioned concurrency configurations for a function
Returns a function’s tags
Returns a list of versions, with the version-specific configuration of each
Creates an Lambda layer from a ZIP archive
Creates a version from the current code and configuration of a function
Update the code signing configuration for the function
Sets the maximum number of simultaneous executions for a function, and reserves capacity for that concurrency level
Configures options for asynchronous invocation on a function, version, or alias
Adds a provisioned concurrency configuration to a function’s alias or version
Sets the runtime management configuration for a function’s version
Removes a statement from the permissions policy for a version of an Lambda layer
remove_permission

Amazon Lex Model Building Service

Description

Amazon Lex is an AWS service for building conversational voice and text interfaces. Use these actions to create, update, and delete conversational bots for new and existing client applications.

Usage

lexmodelbuildingservice(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
Arguments

config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to True to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
    • profile: The name of a profile to use. If not given, then the default profile is used.
    • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```
svc <- lexmodelbuildingservice(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
```
secret_access_key = "string",
session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string",
close_connection = "logical",
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
creds = list(
  access_key_id = "string",
  secret_access_key = "string",
  session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

create_bot_version Creates a new version of the bot based on the $LATEST version
create_intent_version Creates a new version of an intent based on the $LATEST version of the intent
create_slot_type_version Creates a new version of a slot type based on the $LATEST version of the specified slot type
delete_bot Deletes all versions of the bot, including the $LATEST version
delete_bot_alias Deletes an alias for the specified bot
delete_bot_channel_association Deletes the association between an Amazon Lex bot and a messaging platform
delete_bot_version Deletes a specific version of a bot
delete_intent Deletes all versions of the intent, including the $LATEST version
delete_intent_version Deletes a specific version of an intent
delete_slot_type Deletes all versions of the slot type, including the $LATEST version
delete_slot_type_version Deletes a specific version of a slot type
delete_utterances Deletes stored utterances
get_bot Returns metadata information for a specific bot
get_bot_alias Returns information about an Amazon Lex bot alias
get_bot_aliases Returns a list of aliases for a specified Amazon Lex bot
get_bot_channel_association Returns information about the association between an Amazon Lex bot and a messaging platform
get_bot_channel_associations Returns a list of all of the channels associated with the specified bot
get_bots Returns bot information as follows:
get_bot_versions Gets information about all of the versions of a bot
get_builtin_intent Returns information about a built-in intent
lexmodelsv2

get_builtin_intents
get_builtin_slot_types
get_export
get_import
get_intent
get_intents
get_intent_versions
get_migration
get_migrations
get_slot_type
get_slot_types
get_slot_type_versions
get_utterances_view
list_tags_for_resource
put_bot
put_bot_alias
put_intent
put_slot_type
start_import
start_migration
tag_resource
untag_resource

gets a list of built-in intents that meet the specified criteria
gets a list of built-in slot types that meet the specified criteria
exports the contents of a Amazon Lex resource in a specified format
gets information about an import job started with the StartImport operation
returns information about an intent
returns intent information as follows:
gets information about all of the versions of an intent
provides details about an ongoing or complete migration from an Amazon Lex V1 bot to an Amazon Lex V2 bot
gets a list of migrations between Amazon Lex V1 and Amazon Lex V2
returns information about a specific version of a slot type
returns slot type information as follows:
gets information about all versions of a slot type
use the GetUtterancesView operation to get information about the utterances that your users have made to your bot
gets a list of tags associated with the specified resource
creates an Amazon Lex conversational bot or replaces an existing bot
creates an alias for the specified version of the bot or replaces an alias for the specified bot
creates an intent or replaces an existing intent
creates a custom slot type or replaces an existing custom slot type
starts a job to import a resource to Amazon Lex
starts migrating a bot from Amazon Lex V1 to Amazon Lex V2
adds the specified tags to the specified resource
removes tags from a bot, bot alias or bot channel

Examples

## Not run:
svc <- lexmodelbuildingservice()

# This example shows how to get configuration information for a bot.
svc$get_bot(
    name = "DocOrderPizza",
    versionOrAlias = "$LATEST"
)

## End(Not run)
Usage

lexmodelsv2(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```
svc <- lexmodelsv2(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `batch_create_custom_vocabulary_item` Create a batch of custom vocabulary items for a given bot locale’s custom vocabulary
- `batch_delete_custom_vocabulary_item` Delete a batch of custom vocabulary items for a given bot locale’s custom vocabulary
- `batch_update_custom_vocabulary_item` Update a batch of custom vocabulary items for a given bot locale’s custom vocabulary
- `build_bot_locale` Builds a bot, its intents, and its slot types into a specific locale
- `create_bot` Creates an Amazon Lex conversational bot
- `create_bot_alias` Creates an alias for the specified version of a bot
- `create_bot_locale` Creates a locale in the bot
- `create_bot_replica` Action to create a replication of the source bot in the secondary region
- `create_bot_version` Creates an immutable version of the bot
- `create_export` Creates a zip archive containing the contents of a bot or a bot locale
- `create_intent` Creates an intent
- `create_resource_policy` Creates a new resource policy with the specified policy statements
- `create_resource_policy_statement` Adds a new resource policy statement to a bot or bot alias
- `create_slot` Creates a slot in an intent
create_slot_type
create_test_set_discrepancy_report
create_upload_url
delete_bot
delete_bot_alias
delete_bot_locale
delete_bot_replica
delete_bot_version
delete_custom_vocabulary
delete_export
delete_import
delete_intent
delete_resource_policy
delete_resource_policy_statement
delete_slot
delete_slot_type
delete_test_set
delete_utterances
describe_bot
describe_bot_alias
describe_bot_locale
describe_bot_recommendation
describe_bot_replica
describe_bot_resource_generation
describe_bot_version
describe_custom_vocabulary_metadata
describe_export
describe_import
describe_intent
describe_resource_policy
describe_slot
describe_slot_type
describe_test_execution
describe_test_set
describe_test_set_discrepancy_report
describe_test_set_generation
generate_bot_element
generate_test_execution_artifacts_url
list_aggregated_utterances
list_bot_aliases
list_bot_alias_replicas
list_bot_locales
list_bot_recommendations
list_bot_replicas
list_bot_resource_generations
list_bots
list_bot_version_replicas
list_bot_versions

create_slot_type
Create a custom slot type
create_test_set_discrepancy_report
Create a report that describes the differences between the bot and the test set
create_upload_url
Gets a pre-signed S3 write URL that you use to upload the zip archive when importing a bot
delete_bot
Deletes all versions of a bot, including the Draft version
delete_bot_alias
Deletes the specified bot alias
delete_bot_locale
Removes a locale from a bot
delete_bot_replica
The action to delete the replicated bot in the secondary region
delete_bot_version
Deletes a specific version of a bot
delete_custom_vocabulary
Removes a custom vocabulary from the specified locale in the specified bot
delete_export
Removes a previous export and the associated files stored in an S3 bucket
delete_import
Removes a previous import and the associated file stored in an S3 bucket
delete_intent
Removes the specified intent
delete_resource_policy
Removes an existing policy from a bot or bot alias
delete_resource_policy_statement
Deletes a policy statement from a resource policy
delete_slot
Deletes the specified slot from an intent
delete_slot_type
Deletes a slot type from a bot locale
delete_test_set
The action to delete the selected test set
delete_utterances
Deletes stored utterances
describe_bot
Provides metadata information about a bot
describe_bot_alias
Get information about a specific bot alias
describe_bot_locale
Describes the settings that a bot has for a specific locale
describe_bot_recommendation
Provides metadata information about a bot recommendation
describe_bot_replica
Monitors the bot replication status through the UI console
describe_bot_resource_generation
Returns information about a request to generate a bot through natural language description
describe_bot_version
Provides metadata about a version of a bot
describe_custom_vocabulary_metadata
Provides metadata information about a custom vocabulary
describe_export
Gets information about a specific export
describe_import
Gets information about a specific import
describe_intent
Returns metadata about an intent
describe_resource_policy
Gets the resource policy and policy revision for a bot or bot alias
describe_slot
Gets metadata information about a slot
describe_slot_type
Gets metadata information about a slot type
describe_test_execution
Gets metadata information about the test execution
describe_test_set
Gets metadata information about the test set
describe_test_set_discrepancy_report
Gets metadata information about the test set discrepancy report
describe_test_set_generation
Gets metadata information about the test set generation
generate_bot_element
Generates sample utterances for an intent
generate_test_execution_artifacts_url
The pre-signed Amazon S3 URL to download the test execution result artifacts
list_aggregated_utterances
Provides a list of utterances that users have sent to the bot
list_bot_aliases
Gets a list of aliases for the specified bot
list_bot_alias_replicas
The action to list the replicated bots created from the source bot alias
list_bot_locales
Gets a list of locales for the specified bot
list_bot_recommendations
Get a list of bot recommendations that meet the specified criteria
list_bot_replicas
The action to list the replicated bots
list_bot_resource_generations
Lists the generation requests made for a bot locale
list_bots
 Gets a list of available bots
list_bot_version_replicas
Contains information about all the versions replication statuses applicable for Global Resiliency
list_bot_versions
Gets information about all of the versions of a bot
list_built_in_intents
list_built_in_slot_types
list_custom_vocabulary_items
list_exports
list_imports
list_intent_metrics
list_intent_paths
list_intents
list_intent_stage_metrics
list_recommended_intents
list_session_analytics_data
list_session_metrics
list_slots
list_slot_types
list_tags_for_resource
list_test_execution_result_items
list_test_executions
list_test_set_records
list_test_sets
list_utterance_analytics_data
list_utterance_metrics
search_associated_transcripts
start_bot_recommendation
start_bot_resource_generation
start_import
start_test_execution
start_test_set_generation
stop_bot_recommendation
tag_resource
untag_resource
update_bot
update_bot_alias
update_bot_locale
update_bot_recommendation
update_export
update_intent
update_resource_policy
update_slot
update_slot_type
update_test_set

Gets a list of built-in intents provided by Amazon Lex that you can use in your bot
Gets a list of built-in slot types that meet the specified criteria
Paginated list of custom vocabulary items for a given bot locale’s custom vocabulary
Lists the exports for a bot, bot locale, or custom vocabulary
Lists the imports for a bot, bot locale, or custom vocabulary
Retrieves summary metrics for the intents in your bot
Retrieves summary statistics for a path of intents that users take over sessions with your bot
Get a list of intents that meet the specified criteria
Retrieves summary metrics for the stages within intents in your bot
Gets a list of recommended intents provided by the bot recommendation that you can use in your bot
Retrieves a list of metadata for individual user sessions with your bot
Retrieves summary metrics for the user sessions with your bot
Gets a list of slots that match the specified criteria
Gets a list of slot types that match the specified criteria
Gets a list of tags associated with a resource
Gets a list of test execution result items
The list of test set executions
The list of test set records
The list of the test sets
To use this API operation, your IAM role must have permissions to perform the ListExports operation.
To use this API operation, your IAM role must have permissions to perform the ListImports operation.
Search for associated transcripts that meet the specified criteria
Use this to provide your transcript data, and to start the bot recommendation process.
Starts a request for the descriptive bot builder to generate a bot locale configuration based on the prompt you provide. This generates the configuration for your bot.
Starts importing a bot, bot locale, or custom vocabulary from a zip archive that you uploaded to an S3 bucket.
The action to start test set execution
The action to start the generation of test set
Stop an already running Bot Recommendation request
Adds the specified tags to the specified resource
Removes tags from a bot, bot alias, or bot channel
Updates the configuration of an existing bot
Updates the configuration of an existing bot alias
Updates the settings that a bot has for a specific locale
Updates an existing bot recommendation request
Updates the password used to protect an export zip archive
Updates the settings for an intent
Replaces the existing resource policy for a bot or bot alias with a new one
Updates the settings for a slot
Updates the configuration of an existing slot type
The action to update the test set

Examples

## Not run:
svc <- lexmodelsv2()
svc$batch_create_custom_vocabulary_item(
  Foo = 123
lexruntimeservice

Description

Amazon Lex provides both build and runtime endpoints. Each endpoint provides a set of operations (API). Your conversational bot uses the runtime API to understand user utterances (user input text or voice). For example, suppose a user says "I want pizza", your bot sends this input to Amazon Lex using the runtime API. Amazon Lex recognizes that the user request is for the OrderPizza intent (one of the intents defined in the bot). Then Amazon Lex engages in user conversation on behalf of the bot to elicit required information (slot values, such as pizza size and crust type), and then performs fulfillment activity (that you configured when you created the bot). You use the build-time API to create and manage your Amazon Lex bot. For a list of build-time operations, see the build-time API.

Usage

```r
lexruntimeservice(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
lexruntimeservice

- **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials** Optional credentials shorthand for the config parameter

- **creds**:  
  - **access_key_id**: AWS access key ID  
  - **secret_access_key**: AWS secret access key  
  - **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```
svc <- lexruntimeservice(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    )
  )
)
```
lexruntimev2

Amazon Lex Runtime V2

Description

This section contains documentation for the Amazon Lex V2 Runtime V2 API operations.

Usage

```r
lexruntimev2(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Operations

- **delete_session**: Removes session information for a specified bot, alias, and user ID
- **get_session**: Returns session information for a specified bot, alias, and user ID
- **post_content**: Sends user input (text or speech) to Amazon Lex
- **post_text**: Sends user input to Amazon Lex
- **put_session**: Creates a new session or modifies an existing session with an Amazon Lex bot

Examples

```r
## Not run:
svc <- lexruntimeservice()
svc$delete_session(
  Foo = 123
)

## End(Not run)
```
Arguments

config Optional configuration of credentials, endpoint, and/or region.

- credentials:
  - creds:
    * access_key_id: AWS access key ID
    * secret_access_key: AWS secret access key
    * session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.

- endpoint: The complete URL to use for the constructed client.

- region: The AWS Region used in instantiating the client.

- close_connection: Immediately close all HTTP connections.

- timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

- s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

- sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter

- creds:
  - access_key_id: AWS access key ID
  - secret_access_key: AWS secret access key
  - session_token: AWS temporary session token

- profile: The name of a profile to use. If not given, then the default profile is used.

- anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- lexruntimev2(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
```
secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
),
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

**delete_session**    Removes session information for a specified bot, alias, and user ID
**get_session**      Returns session information for a specified bot, alias, and user
**put_session**      Creates a new session or modifies an existing session with an Amazon Lex V2 bot
**recognize_text**   Sends user input to Amazon Lex V2
**recognize_utterance** Sends user input to Amazon Lex V2

Examples

```r
## Not run:
svc <- lexruntimev2()
svc$delete_session(
    Foo = 123
)

## End(Not run)
```
License Manager makes it easier to manage licenses from software vendors across multiple Amazon Web Services accounts and on-premises servers.

Usage

```python
licensemanager(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

Arguments

- `config`: Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
    - `creds`:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
    - `anonymous`: Set anonymous credentials.
  - `endpoint`: The complete URL to use for the constructed client.
  - `region`: The AWS Region used in instantiating the client.
  - `close_connection`: Immediately close all HTTP connections.
  - `timeout`: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - `s3_force_path_style`: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.

- `credentials`: Optional credentials shorthand for the `config` parameter
  - `creds`:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
licensemanager

- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**
A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- licensemanager(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

**Operations**

- **accept_grant**: Accepts the specified grant
<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>check_in_license</td>
<td>Checks the specified license</td>
</tr>
<tr>
<td>checkout_borrow_license</td>
<td>Checks out the specified license for offline use</td>
</tr>
<tr>
<td>checkout_license</td>
<td>Checks out the specified license</td>
</tr>
<tr>
<td>create_grant</td>
<td>Creates a grant for the specified license</td>
</tr>
<tr>
<td>create_grant_version</td>
<td>Creates a new version of the specified grant</td>
</tr>
<tr>
<td>create_license</td>
<td>Creates a license</td>
</tr>
<tr>
<td>create_license_configuration</td>
<td>Creates a license configuration</td>
</tr>
<tr>
<td>create_license_conversion_task_for_resource</td>
<td>Creates a new license conversion task</td>
</tr>
<tr>
<td>create_license_manager_report_generator</td>
<td>Creates a report generator</td>
</tr>
<tr>
<td>create_license_version</td>
<td>Creates a new version of the specified license</td>
</tr>
<tr>
<td>create_token</td>
<td>Creates a long-lived token</td>
</tr>
<tr>
<td>delete_grant</td>
<td>Deletes the specified grant</td>
</tr>
<tr>
<td>delete_license</td>
<td>Deletes the specified license</td>
</tr>
<tr>
<td>delete_license_configuration</td>
<td>Deletes the specified license configuration</td>
</tr>
<tr>
<td>delete_license_manager_report_generator</td>
<td>Deletes the specified report generator</td>
</tr>
<tr>
<td>delete_token</td>
<td>Deletes the specified token</td>
</tr>
<tr>
<td>extend_license_consumption</td>
<td>Extends the expiration date for license consumption</td>
</tr>
<tr>
<td>get_access_token</td>
<td>Gets a temporary access token to use with AssumeRoleWithWebIdentity</td>
</tr>
<tr>
<td>get_grant</td>
<td>Gets detailed information about the specified grant</td>
</tr>
<tr>
<td>get_license</td>
<td>Gets detailed information about the specified license</td>
</tr>
<tr>
<td>get_license_configuration</td>
<td>Gets detailed information about the specified license configuration</td>
</tr>
<tr>
<td>get_license_conversion_task</td>
<td>Gets information about the specified license type conversion task</td>
</tr>
<tr>
<td>get_license_manager_report_generator</td>
<td>Gets information about the specified report generator</td>
</tr>
<tr>
<td>get_license_usage</td>
<td>Gets detailed information about the usage of the specified license</td>
</tr>
<tr>
<td>get_service_settings</td>
<td>Gets the License Manager settings for the current Region</td>
</tr>
<tr>
<td>list_associations_for_license_configuration</td>
<td>Lists the resource associations for the specified license configuration</td>
</tr>
<tr>
<td>list_distributed_grants</td>
<td>Lists the grants distributed for the specified license</td>
</tr>
<tr>
<td>list_failures_for_license_configuration_operations</td>
<td>Lists the license configuration operations that failed</td>
</tr>
<tr>
<td>list_license_configurations</td>
<td>Lists the license configurations for your account</td>
</tr>
<tr>
<td>list_license_conversion_tasks</td>
<td>Lists the license type conversion tasks for your account</td>
</tr>
<tr>
<td>list_license_manager_report_generators</td>
<td>Lists the report generators for your account</td>
</tr>
<tr>
<td>list_licenses</td>
<td>Lists the licenses for your account</td>
</tr>
<tr>
<td>list_license_specifications_for_resource</td>
<td>Describes the license configurations for the specified resource</td>
</tr>
<tr>
<td>list_license_versions</td>
<td>Lists all versions of the specified license</td>
</tr>
<tr>
<td>list_received_grants</td>
<td>Lists grants that are received</td>
</tr>
<tr>
<td>list_received_grants_for_organization</td>
<td>Lists the grants received for all accounts in the organization</td>
</tr>
<tr>
<td>list_received_licenses</td>
<td>Lists received licenses</td>
</tr>
<tr>
<td>list_received_licenses_for_organization</td>
<td>Lists the licenses received for all accounts in the organization</td>
</tr>
<tr>
<td>list_resource_inventory</td>
<td>Lists resources managed using Systems Manager inventory</td>
</tr>
<tr>
<td>list_tags_for_resource</td>
<td>Lists the tags for the specified license configuration</td>
</tr>
<tr>
<td>list_tags_for_resource</td>
<td>Lists your tokens</td>
</tr>
<tr>
<td>list_tokens</td>
<td>Lists all license usage records for a license configuration, displaying license usage records for a license configuration</td>
</tr>
<tr>
<td>reject_grant</td>
<td>Rejects the specified grant</td>
</tr>
<tr>
<td>tag_resource</td>
<td>Adds the specified tags to the specified license configuration</td>
</tr>
<tr>
<td>untag_resource</td>
<td>Removes the specified tags from the specified license configuration</td>
</tr>
<tr>
<td>update_license_configuration</td>
<td>Modifies the attributes of an existing license configuration</td>
</tr>
<tr>
<td>update_license_manager_report_generator</td>
<td>Updates a report generator</td>
</tr>
<tr>
<td>update_license_specifications_for_resource</td>
<td>Adds or removes the specified license configurations for the specified Amazon Web Services resource</td>
</tr>
</tbody>
</table>


Examples

```r
## Not run:
svc <- licensemanager()
svc$accept_grant(
  Foo = 123
)

## End(Not run)
```

## Description

With License Manager, you can discover and track your commercial Linux subscriptions on running Amazon EC2 instances.

## Usage

```r
licensemanagerlinuxsubscriptions(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

## Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
• **close_connection**: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

**credentials** Optional credentials shorthand for the config parameter
- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
csvc <- licensemanagerlinuxsubscriptions(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
```

creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

get_service_settings  Lists the Linux subscriptions service settings
list_linux_subscription_instances  Lists the running Amazon EC2 instances that were discovered with commercial Linux subscriptions
list_linux_subscriptions  Lists the Linux subscriptions that have been discovered
update_service_settings  Updates the service settings for Linux subscriptions

Examples

## Not run:
svc <- licensemanagerlinuxsubscriptions()
svc$get_service_settings(
    Foo = 123
)
## End(Not run)

Description

With License Manager, you can create user-based subscriptions to utilize licensed software with a per user subscription fee on Amazon EC2 instances.

Usage

licensemanagerusersubscriptions(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
Arguments

**config** Optional configuration of credentials, endpoint, and/or region.

  - **credentials**:
    - **creds**:
      * **access_key_id**: AWS access key ID
      * **secret_access_key**: AWS secret access key
      * **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials** Optional credentials shorthand for the config parameter

  - **creds**:
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
    - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- licensemanagerusersubscriptions(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string",
      ),
      profile = "string",
      anonymous = FALSE
    )
  )
)```

secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
),
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

associate_user Associates the user to an EC2 instance to utilize user-based subscriptions
deregister_identity_provider Deregisters the identity provider from providing user-based subscriptions
disassociate_user Disassociates the user from an EC2 instance providing user-based subscriptions
list_identity_providers Lists the identity providers for user-based subscriptions
list_instances Lists the EC2 instances providing user-based subscriptions
list_product_subscriptions Lists the user-based subscription products available from an identity provider
list_user_associations Lists user associations for an identity provider
register_identity_provider Registers an identity provider for user-based subscriptions
start_product_subscription Starts a product subscription for a user with the specified identity provider
stop_product_subscription Stops a product subscription for a user with the specified identity provider
update_identity_provider_settings Updates additional product configuration settings for the registered identity provider

Examples

## Not run:
svc <- licensemanagerusersubscriptions()
svc$associate_user(
    Foo = 123
Amazon Lightsail

Description

Amazon Lightsail is the easiest way to get started with Amazon Web Services (Amazon Web Services) for developers who need to build websites or web applications. It includes everything you need to launch your project quickly - instances (virtual private servers), container services, storage buckets, managed databases, SSD-based block storage, static IP addresses, load balancers, content delivery network (CDN) distributions, DNS management of registered domains, and resource snapshots (backups) - for a low, predictable monthly price.

You can manage your Lightsail resources using the Lightsail console, Lightsail API, Command Line Interface (CLI), or SDKs. For more information about Lightsail concepts and tasks, see the Amazon Lightsail Developer Guide.

This API Reference provides detailed information about the actions, data types, parameters, and errors of the Lightsail service. For more information about the supported Amazon Web Services Regions, endpoints, and service quotas of the Lightsail service, see Amazon Lightsail Endpoints and Quotas in the Amazon Web Services General Reference.

Usage

```r
lightsail(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
• **region**: The AWS Region used in instantiating the client.
• **close_connection**: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

**credentials** Optional credentials shorthand for the config parameter

• **creds**:
  – **access_key_id**: AWS access key ID
  – **secret_access_key**: AWS secret access key
  – **session_token**: AWS temporary session token
• **profile**: The name of a profile to use. If not given, then the default profile is used.
• **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- lightsail(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
)```

credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

allocate_static_ip
attach_certificate_to_distribution
attach_disk
attach_instances_to_load_balancer
attach_load_balancer_tls_certificate
attach_static_ip
close_instance_public_ports
copy_snapshot
create_bucket
create_bucket_access_key
create_certificate
create_cloud_formation_stack
create_contact_method
create_container_service
create_container_service_deployment
create_container_service_registry_login
create_disk
create_disk_from_snapshot
create_disk_snapshot
create_distribution
create_domain
create_domain_entry
create_gui_session_access_details
create_instances
create_instances_from_snapshot
create_instance_snapshot
create_key_pair
create_load_balancer
create_load_balancer_tls_certificate
create_relational_database
create_relational_database_from_snapshot
create_relational_database_snapshot
delete_alarm

Allocates a static IP address
Attaches an SSL/TLS certificate to your Amazon Lightsail content delivery network
Attaches a block storage disk to a running or stopped Lightsail instance and exposes it to the instance with the specified disk name
Attaches one or more Lightsail instances to a load balancer
Attaches a Transport Layer Security (TLS) certificate to your load balancer
Attaches a static IP address to a specific Amazon Lightsail instance
Closes ports for a specific Amazon Lightsail instance
Copies a manual snapshot of an instance or disk as another manual snapshot
Creates an Amazon Lightsail bucket
Creates a new access key for the specified Amazon Lightsail bucket
Creates an SSL/TLS certificate for an Amazon Lightsail content delivery network
Creates an AWS CloudFormation stack, which creates a new Amazon EC2 instance
Creates an email or SMS text message contact method
Creates an Amazon Lightsail container service
Creates a deployment for your Amazon Lightsail container service
Creates a temporary set of log in credentials that you can use to log in to the virtual computer
Creates a block storage disk that can be attached to an Amazon Lightsail instance
Creates a block storage disk from a manual or automatic snapshot of a disk
Creates a snapshot of a block storage disk
Creates an Amazon Lightsail content delivery network (CDN) distribution
Creates a domain resource for the specified domain (example)
Creates one of the following domain name system (DNS) records in a domain
Creates two URLs that are used to access a virtual computer’s graphical user interface
Creates one or more Amazon Lightsail instances
Creates one or more new instances from a manual or automatic snapshot of
Creates a snapshot of a specific virtual private server, or instance
Creates a custom SSH key pair that you can use with an Amazon Lightsail instance
Creates a Lightsail load balancer
Creates an SSL/TLS certificate for an Amazon Lightsail load balancer
Creates a new database in Amazon Lightsail
Creates a new database from an existing database snapshot in Amazon Lightsail
Creates a snapshot of your database in Amazon Lightsail
Deletes an alarm
delete_auto_snapshot
delete_bucket
delete_bucket_access_key
delete_certificate
delete_contact_method
delete_container_image
delete_container_service
delete_disk
delete_disk_snapshot
delete_distribution
delete_domain
delete_domain_entry
delete_instance
delete_instance_snapshot
delete_key_pair
delete_known_host_keys
delete_load_balancer
delete_load_balancer_tls_certificate
delete_relational_database
delete_relational_database_snapshot
detach_certificate_from_distribution
detach_disk
detach_instances_from_load_balancer
detach_static_ip
disable_add_on
download_default_key_pair
download_add_on
export_snapshot
get_active_names
get_alarms
get_auto_snapshots
get_blueprints
get_bucket_access_keys
get_bucket_bundles
get_bucket_metric_data
get_buckets
get_bundles
get_certificates
get_cloud_formation_stack_records
get_contact_methods
get_container_api_metadata
get_container_images
get_container_log
get_container_service_deployments
get_container_service_metric_data
get_container_service_powers
get_container_services
get_cost_estimate

Deletes an automatic snapshot of an instance or disk
Deletes a Amazon Lightsail bucket
Deletes an access key for the specified Amazon Lightsail bucket
Deletes an SSL/TLS certificate for your Amazon Lightsail content delivery network
Deletes a contact method
Deletes a container image that is registered to your Amazon Lightsail container service
Deletes your Amazon Lightsail container service
Deletes the specified block storage disk
Deletes the specified disk snapshot
Deletes your Amazon Lightsail content delivery network (CDN) distribution
Deletes the specified domain recordset and all of its domain records
Deletes a specific domain entry
Deletes an Amazon Lightsail instance
Deletes a specific snapshot of a virtual private server (or instance)
Deletes the specified key pair by removing the public key from Amazon Lightsail
Deletes the known host key or certificate used by the Amazon Lightsail browser-based SSH or RDP clients
Deletes a Lightsail load balancer and all its associated SSL/TLS certificates
Deletes an SSL/TLS certificate associated with a Lightsail load balancer
Deletes a database in Amazon Lightsail
Deletes a database snapshot in Amazon Lightsail
Detaches an SSL/TLS certificate from your Amazon Lightsail content delivery network
Detaches a stopped block storage disk from a Lightsail instance
Detaches the specified instances from a Lightsail load balancer
Detaches a static IP from the Amazon Lightsail instance to which it is attached
Disables an add-on for an Amazon Lightsail resource
Downloads the regional Amazon Lightsail default key pair
Enables or modifies an add-on for an Amazon Lightsail resource
Exports an Amazon Lightsail instance or block storage disk snapshot to Amazon S3
Returns the names of all active (not deleted) resources
Returns information about the configured alarms
Returns the available automatic snapshots for an instance or disk
Returns the list of available instance images, or blueprints
Returns the existing access key IDs for the specified Amazon Lightsail bucket
Returns the bundles that you can apply to a Amazon Lightsail bucket
Returns the data points of a specific metric for an Amazon Lightsail bucket
Returns information about one or more Amazon Lightsail buckets
Returns the bundles that you can apply to an Amazon Lightsail instance when
Returns information about one or more Amazon Lightsail SSL/TLS certificates
Returns the CloudFormation stack record created as a result of the create cloud formation stack operation
Returns information about the configured contact methods
Returns information about Amazon Lightsail containers, such as the current
Returns the container images that are registered to your Amazon Lightsail container service
Returns the log events of a container of your Amazon Lightsail container service
Returns the deployments for your Amazon Lightsail container service
Returns the data points of a specific metric of your Amazon Lightsail container service
Returns the list of powers that can be specified for your Amazon Lightsail container service
Returns information about one or more of your Amazon Lightsail container services
Retrieves information about the cost estimate for a specified resource
get_disk
get_disks
get_disk_snapshot
get_disk_snapshots
get_distribution_bundles
get_distribution_latest_cache_reset
get_distribution_metric_data
get_distributions
get_domain
get_domains
get_export_snapshot_records
get_instance
get_instance_access_details
get_instance_metric_data
get_instance_port_states
get_instances
get_instance_snapshot
get_instance_snapshots
get_instance_state
get_key_pair
get_key_pairs
get_load_balancer
get_load_balancer_metric_data
get_load_balancers
get_load_balancer_tls_certificates
get_load_balancer_tls_policies
get_operation
get_operations
get_operations_for_resource
get_regions
get_relational_database
get_relational_database_blueprints
get_relational_database_bundles
get_relational_database_events
get_relational_database_log_events
get_relational_database_log_streams
get_relational_database_master_user_password
get_relational_database_metric_data
get_relational_databases
get_relational_database_snapshot
get_relational_database_snapshots
get_setup_history
get_static_ip
get_static_ips
import_key_pair
is_vpc_peered
open_instance_public_ports

Returns information about a specific block storage disk
Returns information about all block storage disks in your AWS account and region
Returns information about a specific block storage disk snapshot
Returns information about all block storage disk snapshots in your AWS account and region
Returns the bundles that can be applied to your Amazon Lightsail content delivery network (CDN) distributions
Returns the timestamp and status of the last cache reset of a specific Amazon Lightsail content delivery network (CDN) distribution
Returns the data points of a specific metric for an Amazon Lightsail content delivery network (CDN) distribution
Returns information about one or more of your Amazon Lightsail content delivery network (CDN) distributions
Returns a list of all domains in the user’s account
Returns all export snapshot records created as a result of the export snapshot operation
Returns information about a specific Amazon Lightsail instance, which is a virtual private server (VPS)
Returns temporary SSH keys you can use to connect to a specific virtual private server (VPS)
Returns the data points for the specified Amazon Lightsail instance metric, given an instance name
Returns the firewall port states for a specific Amazon Lightsail instance, the IP addresses allowed to connect to the instance through the ports, and the protocol
Returns information about all Amazon Lightsail virtual private servers, or instances
Returns information about a specific instance snapshot
Returns all instance snapshots for the user’s account
Returns the state of a specific instance
Returns information about a specific key pair
Returns information about all key pairs in the user’s account
Returns information about the specified Lightsail load balancer
Returns information about health metrics for your Lightsail load balancers
Returns information about all load balancers in an account
Returns information about the TLS certificates that are associated with the specified Lightsail load balancer
Returns a list of TLS security policies that you can apply to Lightsail load balancers
Returns information about a specific operation
Returns information about all operations
Gets operations for a specific resource (an instance or a static IP)
Returns a list of all valid regions for Amazon Lightsail
Returns information about a specific database in Amazon Lightsail
Returns a list of available database blueprints in Amazon Lightsail
Returns the list of bundles that are available in Amazon Lightsail
Returns a list of events for a specific database in Amazon Lightsail
Returns a list of log events for a database in Amazon Lightsail
Returns a list of available log streams for a specific database in Amazon Lightsail
Returns the current, previous, or pending versions of the master user password
Returns the data points of the specified metric for a database in Amazon Lightsail
Returns all of the runtime parameters offered by the underlying database software
Returns information about all of your databases in Amazon Lightsail
Returns information about a specific database snapshot in Amazon Lightsail
Returns information about all of your database snapshots in Amazon Lightsail
Returns detailed information for five of the most recent SetupInstanceHttps requests that were run on an instance
Returns information about an Amazon Lightsail static IP
Returns information about all static IPs in the user’s account
Imports a public SSH key from a specific key pair
Returns a Boolean value indicating whether your Lightsail VPC is peered
Opens ports for a specific Amazon Lightsail instance, and specifies the IP addresses allowed to connect to it.
peer_vpc
put_alarm
put_instance_public_ports
reboot_instance
reboot_relational_database
register_container_image
release_static_ip
reset_distribution_cache
send_contact_method_verification
set_ip_address_type
set_resource_access_for_bucket
setup_instance_https
start_gui_session
start_instance
start_relational_database
stop_gui_session
stop_instance
stop_relational_database
tag_resource
test_alarm
unpeer_vpc
untag_resource
update_bucket
update_bucket_bundle
update_container_service
update_distribution
update_distribution_bundle
update_domain_entry
update_instance_metadata_options
update_load_balancer_attribute
update_relational_database
update_relational_database_parameters

Peers the Lightsail VPC with the user’s default VPC
Creates or updates an alarm, and associates it with the specified metric
Opens ports for a specific Amazon Lightsail instance, and specifies the IP addresses allowed to connect to the instance through the ports, and the protocol
Restarts a specific instance
Restarts a specific database in Amazon Lightsail
Registers a container image to your Amazon Lightsail container service
Deletes a specific static IP from your account
Deletes currently cached content from your Amazon Lightsail content delivery network (CDN)
Initiates a graphical user interface (GUI) session that’s used to access a virtual computer’s operating system or application
Starts a specific Amazon Lightsail instance from a stopped state
Starts a specific database from a stopped state in Amazon Lightsail
Terminates a web-based NICE DCV session that’s used to access a virtual computer’s operating system or application
Starts a specific Amazon Lightsail instance that is currently running
Stops a specific database that is currently running in Amazon Lightsail
Adds one or more tags to the specified Amazon Lightsail resource
Tests an alarm by displaying a banner on the Amazon Lightsail console
Unpeers the Lightsail VPC from the user’s default VPC
Deletes the specified set of tag keys and their values from the specified Amazon Lightsail resource
Creates an SSL/TLS certificate that secures traffic for your website
Starts a specific Amazon Lightsail instance from a stopped state
Starts a specific database from a stopped state in Amazon Lightsail
Initiates a graphical user interface (GUI) session that’s used to access a virtual computer’s operating system or application
Starts a specific Amazon Lightsail instance from a stopped state
Starts a specific database from a stopped state in Amazon Lightsail
Unpeers the Lightsail VPC from the user’s default VPC
Updates an existing Amazon Lightsail bucket
Updates the bundle, or storage plan, of an existing Amazon Lightsail bucket
Updates the configuration of your Amazon Lightsail container service, such as your domain name
Updates an existing Amazon Lightsail content delivery network (CDN) distribution
Updates the bundle of your Amazon Lightsail content delivery network (CDN) distribution
Updates a domain recordset after it is created
Updates the Amazon Lightsail instance metadata parameters on a running or stopped instance
Updates one or more attributes of a database in Amazon Lightsail
Updates the specified attribute for a load balancer
Allows the update of one or more attributes of a database in Amazon Lightsail
Allows the update of one or more parameters of a database in Amazon Lightsail

Examples

```r
## Not run:
svc <- lightsail()
svc$allocate_static_ip(
  Foo = 123
)
## End(Not run)
```
Amazon Location Service

Description

"Suite of geospatial services including Maps, Places, Routes, Tracking, and Geofencing"

Usage

locationservice(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)

Arguments

config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoint.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
• **anonymous**: Set anonymous credentials.

```r
svc <- locationservice(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- locationservice(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

**Operations**

- `associate_tracker_consumer`  Creates an association between a geofence collection and a tracker resource
- `batch_delete_device_position_history`  Deletes the position history of one or more devices from a tracker resource
- `batch_delete_geofence`  Deletes a batch of geofences from a geofence collection
batch_evaluate_geofences
batch_get_device_position
batch_put_geofence
batch_update_device_position
calculate_route
calculate_route_matrix
create_geofence_collection
create_key
create_map
create_place_index
create_route_calculator
create_tracker
delete_geofence_collection
delete_key
delete_map
delete_place_index
delete_route_calculator
delete_tracker
describe_geofence_collection
describe_key
describe_map
describe_place_index
describe_route_calculator
describe_tracker
disassociate_tracker_consumer
descr...
untag_resource  Remove one or more tags from the specified Amazon Location resource
update_geofence_collection  Updates the specified properties of a given geofence collection
update_key  Updates the specified properties of a given API key resource
update_map  Updates the specified properties of a given map resource
update_place_index  Updates the specified properties of a given place index resource
update_route_calculator  Updates the specified properties for a given route calculator resource
update_tracker  Updates the specified properties of a given tracker resource

Examples

```
## Not run:
svc <- locationservice()
svc$associate_tracker_consumer(
  Foo = 123
)
## End(Not run)
```

lookoutequipment  Amazon Lookout for Equipment

Description

Amazon Lookout for Equipment is a machine learning service that uses advanced analytics to identify anomalies in machines from sensor data for use in predictive maintenance.

Usage

```
lookoutequipment(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- config  Optional configuration of credentials, endpoint, and/or region.
  - credentials:
    - creds:
      - access_key_id: AWS access key ID
      - secret_access_key: AWS secret access key
      - session_token: AWS temporary session token
– **profile**: The name of a profile to use. If not given, then the default profile is used.
– **anonymous**: Set anonymous credentials.

- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

**credentials** Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- lookoutequipment(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
```

timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

create_dataset
create_inference_scheduler
create_label
create_label_group
create_model
create_retraining_scheduler
delete_dataset
delete_inference_scheduler
delete_label
delete_label_group
delete_model
delete_resource_policy
delete_retraining_scheduler
describe_data_ingestion_job
describe_dataset
describe_inference_scheduler
describe_label
describe_label_group
describe_model
describe_model_version
describe_resource_policy
describe_retraining_scheduler
import_dataset
import_model_version
list_data_ingestion_jobs
list_datasets
list_inference_events
list_inference_executions
list_inferenceSchedulers

Creates a container for a collection of data being ingested for analysis
Creates a scheduled inference
Creates a label for an event
Creates a group of labels
Creates a machine learning model for data inference
Creates a retraining scheduler on the specified model
Deletes a dataset and associated artifacts
Deletes an inference scheduler that has been set up
Deletes a label
Deletes a group of labels
Deletes a machine learning model currently available for Amazon Lookout for Equipment
Deletes the resource policy attached to the resource
Deletes a retraining scheduler from a model
Provides information on a specific data ingestion job such as creation time, dataset ARN, and
Provides a JSON description of the data in each time series dataset, including names, column
Specifies information about the inference scheduler being used, including name, model, status
Returns the name of the label
Returns information about the label group
Provides a JSON containing the overall information about a specific machine learning model,
Retrieves information about a specific machine learning model version
Provides the details of a resource policy attached to a resource
Provides a description of the retraining scheduler, including information such as the model name
Imports a dataset
Imports a model that has been trained successfully
Provides a list of all data ingestion jobs, including dataset name and ARN, S3 location of the
Lists all datasets currently available in your account, filtering on the dataset name
Lists all inference events that have been found for the specified inference scheduler
Lists all inference executions that have been performed by the specified inference scheduler
Retrieves a list of all inference schedulers currently available for your account
### list_label_groups
Returns a list of the label groups

### list_labels
Provides a list of labels

### list_models
Generates a list of all models in the account, including model name and ARN, dataset, and status

### list_model_versions
Generates a list of all model versions for a given model, including the model version, model version ARN, and status

### list_retraining_schedulers
Lists all retraining schedulers in your account, filtering by model name prefix and status

### list_sensor_statistics
Lists statistics about the data collected for each of the sensors that have been successfully ingested

### list_tags_for_resource
Lists all the tags for a specified resource, including key and value

### put_resource_policy
Creates a resource control policy for a given resource

### start_data_ingestion_job
Starts a data ingestion job

### start_inference_scheduler
Starts an inference scheduler

### start_retraining_scheduler
Starts a retraining scheduler

### stop_inference_scheduler
Stops an inference scheduler

### stop_retraining_scheduler
Stops a retraining scheduler

### tag_resource
Associates a given tag to a resource in your account

### untag_resource
Removes a specific tag from a given resource

### update_active_model_version
Sets the active model version for a given machine learning model

### update_inference_scheduler
Updates an inference scheduler

### update_label_group
Updates the label group

### update_model
Updates a model in the account

### update_retraining_scheduler
Updates a retraining scheduler

### Examples

```r
## Not run:
svc <- lookoutequipment()
#
svc$create_retraining_scheduler(
  ClientToken = "sample-client-token",
  LookbackWindow = "P360D",
  ModelName = "sample-model",
  PromoteMode = "MANUAL",
  RetrainingFrequency = "P1M"
)
## End(Not run)
```

---

**Description**

This is the *Amazon Lookout for Metrics API Reference*. For an introduction to the service with tutorials for getting started, visit *Amazon Lookout for Metrics Developer Guide*. 
Usage

lookoutmetrics(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

config

  Optional configuration of credentials, endpoint, and/or region.

  - credentials:
    - creds:
      - access_key_id: AWS access key ID
      - secret_access_key: AWS secret access key
      - session_token: AWS temporary session token
    - profile: The name of a profile to use. If not given, then the default profile is used.
    - anonymous: Set anonymous credentials.

  endpoint: The complete URL to use for the constructed client.

  region: The AWS Region used in instantiating the client.

  close_connection: Immediately close all HTTP connections.

  timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

  s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

  sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ep.html

credentials

  Optional credentials shorthand for the config parameter

  - creds:
    - access_key_id: AWS access key ID
    - secret_access_key: AWS secret access key
    - session_token: AWS temporary session token

  profile: The name of a profile to use. If not given, then the default profile is used.

  anonymous: Set anonymous credentials.

endpoint

  Optional shorthand for complete URL to use for the constructed client.

region

  Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```
svc <- lookoutmetrics(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `activate_anomaly_detector`: Activates an anomaly detector
- `back_test_anomaly_detector`: Runs a backtest for anomaly detection for the specified resource
- `create_alert`: Creates an alert for an anomaly detector
- `create_anomaly_detector`: Creates an anomaly detector
- `create_metric_set`: Creates a dataset
- `deactivate_anomaly_detector`: Deactivates an anomaly detector
- `delete_alert`: Deletes an alert
- `delete_anomaly_detector`: Deletes a detector
- `describe_alert`: Describes an alert
- `describe_anomaly_detection_executions`: Returns information about the status of the specified anomaly detection jobs
- `describe_anomaly_detector`: Describes a detector
- `describe_metric_set`: Describes a dataset
- `detect_metric_set_config`: Detects an Amazon S3 dataset’s file format, interval, and offset
- `get_anomaly_group`: Returns details about a group of anomalous metrics
get_data_quality_metrics
get_feedback
get_sample_data
list_alerts
list_anomaly_detectors
list_anomaly_group_related_metrics
list_anomaly_group_summaries
list_anomaly_group_time_series
list_metric_sets
list_tags_for_resource
put_feedback
tag_resource
untag_resource
update_alert
update_anomaly_detector
update_metric_set

Returns details about the requested data quality metrics
Get feedback for an anomaly group
Returns a selection of sample records from an Amazon S3 datasource
Lists the alerts attached to a detector
Lists the detectors in the current AWS Region
Returns a list of measures that are potential causes or effects of an anomaly group
Returns a list of anomaly groups
Gets a list of anomalous metrics for a measure in an anomaly group
Lists the datasets in the current AWS Region
Gets a list of tags for a detector, dataset, or alert
Add feedback for an anomalous metric
Adds tags to a detector, dataset, or alert
Removes tags from a detector, dataset, or alert
Make changes to an existing alert
Updates a detector
Updates a dataset

Examples

```r
## Not run:
svc <- lookoutmetrics()
svc$activate_anomaly_detector(  
  Foo = 123  
)

## End(Not run)
```

---

**machinelearning**

**Amazon Machine Learning**

**Description**

Definition of the public APIs exposed by Amazon Machine Learning

**Usage**

```r
machinelearning(  
  config = list(),  
  credentials = list(),  
  endpoint = NULL,  
  region = NULL  
)
```
Arguments

config Optional configuration of credentials, endpoint, and/or region.

  • credentials:
    - creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    - profile: The name of a profile to use. If not given, then the default
      profile is used.
    - anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when at-
    tempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style
    addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or
    html

credentials Optional credentials shorthand for the config parameter

  • creds:
    - access_key_id: AWS access key ID
    - secret_access_key: AWS secret access key
    - session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile
    is used.
  • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...),
where svc is the name you’ve assigned to the client. The available operations are listed in the Op-
erations section.

Service syntax

svc <- machinelearning(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
    
SVC論文ナビゲーション

  • access_key_id: AWS access key ID
  • secret_access_key: AWS secret access key
  • session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile
    is used.
  • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...),
where svc is the name you’ve assigned to the client. The available operations are listed in the Op-
erations section.

Service syntax

svc <- machinelearning(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
),
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

add_tags
create_batch_prediction
create_data_source_from_rds
create_data_source_from_redshift
create_data_source_from_s3
create_evaluation
create_ml_model
create_realtime_endpoint
delete_batch_prediction
delete_data_source
delete_evaluation
delete_ml_model
delete_realtime_endpoint
delete_tags
describe_batch_predictions
describe_data_sources
describe_evaluations
describe_ml_models
describe_tags
get_batch_prediction

Add one or more tags to an object, up to a limit of 10
Generates predictions for a group of observations
Creates a DataSource object from an Amazon Relational Database Service (Amazon RDS)
Creates a DataSource from a database hosted on an Amazon Redshift cluster
Creates a DataSource object
Creates a new Evaluation of an MLModel
Creates a new MLModel using the DataSource and the recipe as information sources
Assigns the DELETED status to a BatchPrediction, rendering it unusable
Assigns the DELETED status to a DataSource, rendering it unusable
Assigns the DELETED status to an Evaluation, rendering it unusable
Assigns the DELETED status to an MLModel, rendering it unusable
Deletes a real time endpoint of an MLModel
Deletes the specified tags associated with an ML object
Returns a list of BatchPrediction operations that match the search criteria in the request
Returns a list of DataSource that match the search criteria in the request
Returns a list of DescribeEvaluations that match the search criteria in the request
Returns a list of MLModel that match the search criteria in the request
Describes one or more of the tags for your Amazon ML object
Returns a BatchPrediction that includes detailed metadata, status, and data file information
macie2

get_data_source
get_evaluation
get_ml_model
predict
update_batch_prediction
update_data_source
update_evaluation
update_ml_model

Returns a DataSource that includes metadata and data file information, as well as the current status of the DataSource.

Returns an Evaluation that includes metadata as well as the current status of the Evaluation.

Returns an MLModel that includes detailed metadata, data source information, and the current status of the MLModel.

Generates a prediction for the observation using the specified ML Model.

Updates the BatchPredictionName of a BatchPrediction.

Updates the DataSourceName of a DataSource.

Updates the EvaluationName of an Evaluation.

Updates the MLModelName and the ScoreThreshold of an MLModel.

Examples

## Not run:
svc <- machinelearning()
svc$add_tags(
  Foo = 123
)

## End(Not run)

macie2

Amazon Macie 2

Description

Amazon Macie

Usage

macie2(config = list(), credentials = list(), endpoint = NULL, region = NULL)

Arguments

config Optional configuration of credentials, endpoint, and/or region.

• credentials:
  • creds:
    • access_key_id: AWS access key ID
    • secret_access_key: AWS secret access key
    • session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.
• endpoint: The complete URL to use for the constructed client.
• region: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials**
Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**
A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- macie2(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(  
```
creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

accept_invitation
batch_get_custom_data_identifiers
create_allow_list
create_classification_job
create_custom_data_identifier
create_findings_filter
create_invitations
create_member
create_sample_findings
decline_invitations
delete_allow_list
delete_custom_data_identifier
delete_findings_filter
delete_invitations
delete_member
describe_buckets
describe_classification_job
describe_organization_configuration
disable_macie
disable_organization_admin_account
disassociate_from_administrator_account
disassociate_from_master_account
disassociate_member
enable_macie
enable_organization_admin_account
get_administrator_account
get_allow_list
get_automated_discovery_configuration
get_bucket_statistics
get_classification_export_configuration
get_classification_scope
get_custom_data_identifier
get_findings
get_findings_filter

Accepts an Amazon Macie membership invitation that was received from a specific account
Retrieves information about one or more custom data identifiers
Creates and defines the settings for an allow list
Creates and defines the settings for a classification job
Creates and defines the criteria and other settings for a custom data identifier
Creates and defines the criteria and other settings for a findings filter
Sends an Amazon Macie membership invitation to one or more accounts
Associates an account with an Amazon Macie administrator account
Creates sample findings
Declines Amazon Macie membership invitations that were received from specific accounts
Deletes an allow list
Soft deletes a custom data identifier
Deletes a findings filter
Deletes Amazon Macie membership invitations that were received from specific accounts
Deletes the association between an Amazon Macie administrator account and another account
Retrieves (queries) statistical data and other information about one or more S3 buckets
Retrieves the status and settings for a classification job
Retrieves the Amazon Macie configuration settings for an organization in Organizations
Disables Amazon Macie and deletes all settings and resources for a Macie account
Disables an account as the delegated Amazon Macie administrator account for an organization in Organizations
Disassociates a member account from its Amazon Macie administrator account
Disassociates a member account from its Amazon Macie administrator account
Disassociates an Amazon Macie administrator account from a member account
Enables Amazon Macie and specifies the configuration settings for a Macie account
Designates an account as the delegated Amazon Macie administrator account for another account
Retrieves information about the Amazon Macie administrator account for an account
Retrieves the settings and status of an allow list
Retrieves the configuration settings and status of automated sensitive data discovery
Retrieves (queries) aggregated statistical data about all the S3 buckets that Amazon Macie has scanned
Retrieves the configuration settings for storing data classification results
Retrieves the classification scope settings for an account
Retrieves the criteria and other settings for a custom data identifier
Retrieves the details of one or more findings
Retrieves the criteria and other settings for a findings filter
get_findings_publication_configuration
get_finding_statistics
get_invitations_count
get_macie_session
get_master_account
get_member
get_resource_profile
get_reveal_configuration
get_sensitive_data_occurrences
get_sensitive_data_occurrences_availability
get_sensitivity_inspection_template
get_usage_statistics
get_usage_totals
list_allow_lists
list_classification_jobs
list_classification_scopes
list_custom_data_identifiers
list_findings
list_findings_filters
list_invitations
list_managed_data_identifiers
list_members
list_organization_admin_accounts
list_resource_profile_artifacts
list_resource_profile_detections
list_sensitivity_inspection_templates
list_tags_for_resource
put_classification_export_configuration
put_findings_publication_configuration
search_resources
tag_resource
test_custom_data_identifier
untag_resource
update_allow_list
update_automated_discovery_configuration
update_classification_job
update_classification_scope
update_findings_filter
update_macie_session
update_member_session
update_organization_configuration
update_resource_profile
update_resource_profile_detections
update_reveal_configuration
update_sensitivity_inspection_template
Retrieves the configuration settings for publishing findings to Security Hub
Retrieves (queries) aggregated statistical data about findings
Retrieves the count of Amazon Macie membership invitations that were received
Retrieves the status and configuration settings for an Amazon Macie account
(Deprecated) Retrieves information about the Amazon Macie administrator account
Retrieves information about an account that’s associated with an Amazon Macie
Retrieves (queries) sensitive data discovery statistics and the sensitivity score for
Retrieves the status and configuration settings for retrieving occurrences of sensitive
Retrieves occurrences of sensitive data reported by a finding
Checks whether occurrences of sensitive data can be retrieved for a finding
Retrieves the settings for the sensitivity inspection template for an account
Retrieves (queries) quotas and aggregated usage data for one or more accounts
Retrieves (queries) aggregated usage data for an account
Retrieves a subset of information about all the allow lists for an account
Retrieves a subset of information about one or more classification jobs
Retrieves a subset of information about the classification scope for an account
Retrieves a subset of information about all the custom data identifiers for an account
Retrieves a subset of information about one or more findings
Retrieves a subset of information about all the findings filters for an account
Retrieves information about the Amazon Macie membership invitations that were received
Retrieves information about all the managed data identifiers that Amazon Macie
Retrieves information about the accounts that are associated with an Amazon Macie
Retrieves information about the delegated Amazon Macie administrator account
Retrieves information about objects that were selected from an S3 bucket for automated
Retrieves information about the types and amount of sensitive data that Amazon Macie
Retrieves a subset of information about the sensitivity inspection template for an account
Creates or updates the configuration settings for storing data classification results
Updates the configuration settings for publishing findings to Security Hub
Retrieves (queries) statistical data and other information about Amazon Web Services
Adds or updates one or more tags (keys and values) that are associated with an Amazon Macie
Tests a custom data identifier
Removes one or more tags (keys and values) from an Amazon Macie resource
Updates the settings for an allow list
Enables or disables automated sensitive data discovery for an account
Changes the status of a classification job
Updates the classification scope settings for an account
Updates the criteria and other settings for a findings filter
Suspends or re-enables Amazon Macie, or updates the configuration settings for an
Enables an Amazon Macie administrator to suspend or re-enable Macie for a member
Updates the Amazon Macie configuration settings for an organization in Organizations
Updates the sensitivity score for an S3 bucket
Updates the sensitivity scoring settings for an S3 bucket
Updates the status and configuration settings for retrieving occurrences of sensitive
Updates the settings for the sensitivity inspection template for an account
Examples

```r
## Not run:
svc <- macie2()
svc$accept_invitation(
   Foo = 123
)
## End(Not run)
```

---

**managedgrafana**  
*Amazon Managed Grafana*

**Description**

Amazon Managed Grafana is a fully managed and secure data visualization service that you can use to instantly query, correlate, and visualize operational metrics, logs, and traces from multiple sources. Amazon Managed Grafana makes it easy to deploy, operate, and scale Grafana, a widely deployed data visualization tool that is popular for its extensible data support.

With Amazon Managed Grafana, you create logically isolated Grafana servers called *workspaces*. In a workspace, you can create Grafana dashboards and visualizations to analyze your metrics, logs, and traces without having to build, package, or deploy any hardware to run Grafana servers.

**Usage**

```r
managedgrafana(
   config = list(),
   credentials = list(),
   endpoint = NULL,
   region = NULL
)
```

**Arguments**

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - *access_key_id*: AWS access key ID
      - *secret_access_key*: AWS secret access key
      - *session_token*: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
• **close_connection**: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials**
Optional credentials shorthand for the config parameter

• **creds**:
  – **access_key_id**: AWS access key ID
  – **secret_access_key**: AWS secret access key
  – **session_token**: AWS temporary session token
• **profile**: The name of a profile to use. If not given, then the default profile is used.
• **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**
A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- managedgrafana(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
```
creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

## Operations

- **associate_license**
  Assigns a Grafana Enterprise license to a workspace

- **create_workspace**
  Creates a workspace

- **create_workspace_api_key**
  Creates a Grafana API key for the workspace

- **delete_workspace**
  Deletes an Amazon Managed Grafana workspace

- **delete_workspace_api_key**
  Deletes a Grafana API key for the workspace

- **describe_workspace**
  Displays information about one Amazon Managed Grafana workspace

- **describe_workspace_authentication**
  Displays information about the authentication methods used in one Amazon Managed Grafana workspace

- **describe_workspace_configuration**
  Gets the current configuration string for the given workspace

- **disassociate_license**
  Removes the Grafana Enterprise license from a workspace

- **list_permissions**
  Lists the users and groups who have the Grafana Admin and Editor roles in this workspace

- **list_tags_for_resource**
  The ListTagsForResource operation returns the tags that are associated with the Amazon Managed Grafana resource

- **list_versions**
  Lists available versions of Grafana

- **list_workspaces**
  Returns a list of Amazon Managed Grafana workspaces in the account, with some information about each workspace

- **tag_resource**
  The TagResource operation associates tags with an Amazon Managed Grafana resource

- **untag_resource**
  The UntagResource operation removes the association of the tag with the Amazon Managed Grafana resource

- **update_permissions**
  Updates which users in a workspace have the Grafana Admin or Editor roles

- **update_workspace**
  Modifies an existing Amazon Managed Grafana workspace

- **update_workspace_authentication**
  Use this operation to define the identity provider (IdP) that this workspace authenticates users from

- **update_workspace_configuration**
  Updates the configuration string for the given workspace

## Examples

```r
## Not run:
svc <- managedgrafana()
svc$associate_license(
    Foo = 123
)

## End(Not run)
```
marketplacecatalog  AWS Marketplace Catalog Service

Description

Catalog API actions allow you to manage your entities through list, describe, and update capabilities. An entity can be a product or an offer on AWS Marketplace.

You can automate your entity update process by integrating the AWS Marketplace Catalog API with your AWS Marketplace product build or deployment pipelines. You can also create your own applications on top of the Catalog API to manage your products on AWS Marketplace.

Usage

```python
catalog = marketplacecatalog(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

Arguments

config  Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html

credentials  Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
marketplacecatalog

- **secret_access_key**: AWS secret access key
- **session_token**: AWS temporary session token

  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- marketplacecatalog(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```
Operations
marketplacecommerceanalytics

AWS Marketplace Commerce Analytics

Description

Provides AWS Marketplace business intelligence data on-demand.

Usage

marketplacecommerceanalytics(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
Arguments

config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endl.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.
region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```
svc <- marketplacecommerceanalytics(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
```
secret_access_key = "string",
  session_token = "string"
),
  profile = "string",
  anonymous = "logical"
),
  endpoint = "string",
  region = "string",
  close_connection = "logical",
  timeout = "numeric",
  s3_force_path_style = "logical",
  sts_regional_endpoint = "string"
),
credentials = list(
  creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
  ),
  profile = "string",
  anonymous = "logical"
),
  endpoint = "string",
  region = "string"
)

Operations

generate_data_set       Given a data set type and data set publication date, asynchronously publishes the requested data set to the specified S3 bucket and notifies the specified SNS topic once the data is available
start_support_data_export This target has been deprecated

Examples

## Not run:
svc <- marketplacecommerceanalytics()
svc$generate_data_set(
  Foo = 123
)
## End(Not run)
Description

This reference provides descriptions of the AWS Marketplace Entitlement Service API.

AWS Marketplace Entitlement Service is used to determine the entitlement of a customer to a given product. An entitlement represents capacity in a product owned by the customer. For example, a customer might own some number of users or seats in an SaaS application or some amount of data capacity in a multi-tenant database.

Getting Entitlement Records

• GetEntitlements - Gets the entitlements for a Marketplace product.

Usage

marketplaceentitlementservice(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

config Optional configuration of credentials, endpoint, and/or region.

• credentials:
  – creds:
    * access_key_id: AWS access key ID
    * secret_access_key: AWS secret access key
    * session_token: AWS temporary session token
  – profile: The name of a profile to use. If not given, then the default profile is used.
  – anonymous: Set anonymous credentials.

• endpoint: The complete URL to use for the constructed client.

• region: The AWS Region used in instantiating the client.

• close_connection: Immediately close all HTTP connections.

• timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

• s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

• sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html

credentials Optional credentials shorthand for the config parameter

• creds:
  – access_key_id: AWS access key ID
  – secret_access_key: AWS secret access key
  – session_token: AWS temporary session token
profile: The name of a profile to use. If not given, then the default profile is used.

anonymous: Set anonymous credentials.

endpoint: Optional shorthand for complete URL to use for the constructed client.

region: Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- marketplaceentitlements
```

Operations

get_entitlements GetEntitlements retrieves entitlement values for a given product
Examples

```r
## Not run:
svc <- marketplaceentitlementservice()
svc$get_entitlements(
  Foo = 123
)

## End(Not run)
```

---

**marketplacemetering**  
**AWS Marketplace Metering**

**Description**

AWS Marketplace Metering Service

This reference provides descriptions of the low-level AWS Marketplace Metering Service API. AWS Marketplace sellers can use this API to submit usage data for custom usage dimensions.

For information on the permissions you need to use this API, see [AWS Marketplace metering and entitlement API permissions](#) in the [AWS Marketplace Seller Guide](#).

**Submitting Metering Records**

- *MeterUsage* - Submits the metering record for an AWS Marketplace product. `meter_usage` is called from an EC2 instance or a container running on EKS or ECS.

- *BatchMeterUsage* - Submits the metering record for a set of customers. `batch_meter_usage` is called from a software-as-a-service (SaaS) application.

**Accepting New Customers**

- *ResolveCustomer* - Called by a SaaS application during the registration process. When a buyer visits your website during the registration process, the buyer submits a Registration Token through the browser. The Registration Token is resolved through this API to obtain a `CustomerIdentifier` along with the `CustomerAWSAccountId` and `ProductCode`.

**Entitlement and Metering for Paid Container Products**

- Paid container software products sold through AWS Marketplace must integrate with the AWS Marketplace Metering Service and call the `register_usage` operation for software entitlement and metering. Free and BYOL products for Amazon ECS or Amazon EKS aren’t required to call `register_usage`, but you can do so if you want to receive usage data in your seller reports. For more information on using the `register_usage` operation, see [Container-Based Products](#).

batch_meter_usage API calls are captured by AWS CloudTrail. You can use Cloudtrail to verify that the SaaS metering records that you sent are accurate by searching for records with the `eventName` of `batch_meter_usage`. You can also use CloudTrail to audit records over time. For more information, see the [AWS CloudTrail User Guide](#).
Usage

marketplacemetering(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

  config  Optional configuration of credentials, endpoint, and/or region.
          • credentials:
            – creds:
              • access_key_id: AWS access key ID
              • secret_access_key: AWS secret access key
              • session_token: AWS temporary session token
            – profile: The name of a profile to use. If not given, then the default
              profile is used.
            – anonymous: Set anonymous credentials.
          • endpoint: The complete URL to use for the constructed client.
          • region: The AWS Region used in instantiating the client.
          • close_connection: Immediately close all HTTP connections.
          • timeout: The time in seconds till a timeout exception is thrown when at-
            tempting to make a connection. The default is 60 seconds.
          • s3_force_path_style: Set this to true to force the request to use path-style
            addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
          • sts_regional_endpoint: Set sts regional endpoint resolver to regional or

  credentials  Optional credentials shorthand for the config parameter
          • creds:
            – access_key_id: AWS access key ID
            – secret_access_key: AWS secret access key
            – session_token: AWS temporary session token
          • profile: The name of a profile to use. If not given, then the default profile
            is used.
          • anonymous: Set anonymous credentials.

  endpoint  Optional shorthand for complete URL to use for the constructed client.

  region  Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...),
where svc is the name you’ve assigned to the client. The available operations are listed in the Op-
erations section.
Service syntax

```r
csvc <- marketplacemetering(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `batch_meter_usage` BatchMeterUsage is called from a SaaS application listed on AWS Marketplace to post metering records
- `meter_usage` API to emit metering records
- `register_usage` Paid container software products sold through AWS Marketplace must integrate with the AWS Marketplace Metering Service and call the RegisterUsage operation for software entitlement and metering
- `resolve_customer` ResolveCustomer is called by a SaaS application during the registration process

Examples

```r
# Not run:
svc <- marketplacemetering()
svc$batch_meter_usage(
  Foo = 123
)
```
memorydb

Description

MemoryDB for Redis is a fully managed, Redis-compatible, in-memory database that delivers ultra-fast performance and Multi-AZ durability for modern applications built using microservices architectures. MemoryDB stores the entire database in-memory, enabling low latency and high throughput data access. It is compatible with Redis, a popular open source data store, enabling you to leverage Redis’ flexible and friendly data structures, APIs, and commands.

Usage

```r
memorydb(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**: Optional credentials shorthand for the config parameter
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
– **session_token**: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- memorydb(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

**Operations**
batch_update_cluster
    Apply the service update to a list of clusters supplied

copy_snapshot
    Makes a copy of an existing snapshot

create_acl
    Creates an Access Control List

create_cluster
    Creates a cluster

create_parameter_group
    Creates a new MemoryDB parameter group

create_snapshot
    Creates a copy of an entire cluster at a specific moment in time

create_subnet_group
    Creates a subnet group

create_user
    Creates a MemoryDB user

delete_acl
    Deletes an Access Control List

delete_cluster
    Deletes a cluster

delete_parameter_group
    Deletes the specified parameter group

delete_snapshot
    Deletes an existing snapshot

delete_subnet_group
    Deletes a subnet group

delete_user
    Deletes a user

describe_ac_ls
    Returns a list of ACLs

describe_clusters
    Returns information about all provisioned clusters if no cluster identifier is specified, or

describe_engine_versions
    Returns information about all provisioned clusters if no cluster identifier is specified, or

describe_events
    Returns events related to clusters, security groups, and parameter groups

describe_parameter_groups
    Returns a list of parameter group descriptions

describe_parameters
    Returns the detailed parameter list for a particular parameter group

describe_reserved_nodes
    Returns information about reserved nodes for this account, or about a specified reserved

describe_reserved_nodes_offerings
    Lists available reserved node offerings

describe_service_updates
    Returns details of the service updates

describe_snapshots
    Returns information about cluster snapshots

describe_subnet_groups
    Returns a list of subnet group descriptions

describe_users
    Returns a list of users

describe_users
    Returns a list of users

describe_users
    Returns a list of users

describe_subnets
    Returns a list of subnets

describe_subnets
    Returns a list of subnets

describe_subnets
    Returns a list of subnets

describe_subnets
    Returns a list of subnets

describe_subnets
    Returns a list of subnets

describe_subnets
    Returns a list of subnets

describe_subnets
    Returns a list of subnets

describe_subnets
    Returns a list of subnets

describe_subnets
    Returns a list of subnets

describe_subnets
    Returns a list of subnets

describe_subnets
    Returns a list of subnets

describe_subnets
    Returns a list of subnets

describe_subnets
    Returns a list of subnets

describe_subnets
    Returns a list of subnets

describe_subnets
    Returns a list of subnets

describe_subnets
    Returns a list of subnets

describe_subnets
    Returns a list of subnets

describe_subnets
    Returns a list of subnets

Examples

## Not run:
svc <- memorydb()
svc$batch_update_cluster(
    Foo = 123
  )
## Description

Amazon MQ is a managed message broker service for Apache ActiveMQ and RabbitMQ that makes it easy to set up and operate message brokers in the cloud. A message broker allows software applications and components to communicate using various programming languages, operating systems, and formal messaging protocols.

### Usage

```r
mq(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

### Arguments

<table>
<thead>
<tr>
<th>config</th>
<th>Optional configuration of credentials, endpoint, and/or region.</th>
</tr>
</thead>
<tbody>
<tr>
<td>credentials</td>
<td>Optional credentials shorthand for the config parameter</td>
</tr>
</tbody>
</table>

- **config**
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy `https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html`

- **credentials**
  - **creds**:
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
    - **session_token**: AWS temporary session token
mq

- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint**  
Optional shorthand for complete URL to use for the constructed client.

**region**  
Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- mq(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

**Operations**

- **create_broker**: Creates a broker
create_configuration: Creates a new configuration for the specified configuration name
create_tags: Adds a tag to a resource
create_user: Creates an ActiveMQ user
delete_broker: Deletes a broker
delete_tags: Removes a tag from a resource
delete_user: Deletes an ActiveMQ user
describe_broker: Returns information about the specified broker
describe_broker_engine_types: Describe available engine types and versions
describe_broker_instance_options: Describe available broker instance options
describe_configuration: Returns information about the specified configuration
describe_configuration_revision: Returns the specified configuration revision for the specified configuration
describe_user: Returns information about an ActiveMQ user
list_brokers: Returns a list of all brokers
list_configuration_revisions: Returns a list of all revisions for the specified configuration
list_configurations: Returns a list of all configurations
list_tags: Lists tags for a resource
list_users: Returns a list of all ActiveMQ users
promote: Promotes a data replication replica broker to the primary broker role
reboot_broker: Reboots a broker
update_broker: Adds a pending configuration change to a broker
update_configuration: Updates the specified configuration
update_user: Updates the information for an ActiveMQ user

Examples

```r
## Not run:
svc <- mq()
svc$create_broker(
   Foo = 123
)
## End(Not run)
```

mturk

Amazon Mechanical Turk

Description

Amazon Mechanical Turk API Reference

Usage

mturk(config = list(), credentials = list(), endpoint = NULL, region = NULL)
Arguments

config Optional configuration of credentials, endpoint, and/or region.

- credentials:
  - creds:
    - access_key_id: AWS access key ID
    - secret_access_key: AWS secret access key
    - session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.
- endpoint: The complete URL to use for the constructed client.
- region: The AWS Region used in instantiating the client.
- close_connection: Immediately close all HTTP connections.
- timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
- sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter

- creds:
  - access_key_id: AWS access key ID
  - secret_access_key: AWS secret access key
  - session_token: AWS temporary session token
- profile: The name of a profile to use. If not given, then the default profile is used.
- anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```
svc <- mturk(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string",
        profile = "string",
        anonymous = TRUE
      ),
      profile = "string",
      anonymous = TRUE
    ),
    endpoint = "string",
    region = "string",
    close_connection = TRUE,
    timeout = 60,
    s3_force_path_style = TRUE,
    sts_regional_endpoint = "string"
  )
)
```
secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
),
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

accept_qualification_request
approve_assignment
associate_qualification_with_worker
create_additional_assignments_for_hit
create_hit
create_hit_type
create_hit_with_hit_type
create_qualification_type
create_worker_block
delete_hit
delete_qualification_type
delete_worker_block
disassociate_qualification_from_worker
get_account_balance
get_assignment
get_file_upload_url
get_hit
get_qualification_score
get_qualification_type
list_assignments_for_hit

The AcceptQualificationRequest operation approves a Worker’s request for a Qualification.
The ApproveAssignment operation approves the results of a completed assignment.
The AssociateQualificationWithWorker operation gives a Worker a Qualification.
The CreateAdditionalAssignmentsForHIT operation increases the maximum number of assignments of an existing HIT.
The CreateHIT operation creates a new Human Intelligence Task (HIT).
The CreateHITType operation creates a new HIT type.
The CreateHITWithHITType operation creates a new Human Intelligence Task (HIT).
The CreateQualificationType operation creates a new Qualification type, which is required for associate qualifier.
The CreateWorkerBlock operation allows you to prevent a Worker from working on your HITs.
The DeleteHIT operation is used to delete HIT that is no longer needed.
The DeleteQualificationType deletes a Qualification type and deletes any HIT types that use it.
The DeleteWorkerBlock operation allows you to reinstate a blocked Worker to work on your HITs.
The DisassociateQualificationFromWorker revokes a previously granted Qualification.
The GetAccountBalance operation retrieves the Prepaid HITs balance in your Amazon Mechanical Turk (MT) account.
The GetAssignment operation retrieves the details of the specified Assignment.
The GetFileUploadURL operation generates and returns a temporary URL.
The GetHit operation retrieves the details of the specified HIT.
The GetQualificationScore operation returns the value of a Worker’s Qualification for a given Qualification type.
The GetQualificationType operation retrieves information about a Qualification type.
The ListAssignmentsForHIT operation retrieves completed assignments for a HIT.
list_bonusPayments
list_hi_t
list_hi_t_for_qualification_type
list_qualification_requests
list_qualification_types
list_reviewable_hi_t
list_review_policy_results_for_hit
list_worker_blocks
list_workers_with_qualification_type
notify_workers
reject_assignment
reject_qualification_request
send_bonus
send_test_event_notification
update_expiration_for_hit
update_hit_review_status
update_hit_type_of_hit
update_notification_settings
update_qualification_type

Examples

```r
## Not run:
svc <- mturk()
svc$accept_qualification_request(
  Foo = 123
)
## End(Not run)
```

### Description

Amazon Managed Workflows for Apache Airflow

This section contains the Amazon Managed Workflows for Apache Airflow (MWAA) API reference documentation. For more information, see What is Amazon MWAA?

### Endpoints

- api.airflow.{region}.amazonaws.com - This endpoint is used for environment management.
  - create_environment
  - delete_environment

- get_environment
- list_environments
- list_tags_for_resource
- tag_resource
- untag_resource
- update_environment

• env.airflow.{region}.amazonaws.com - This endpoint is used to operate the Airflow environment.
  - create_cli_token
  - create_web_login_token

Regions
For a list of supported regions, see Amazon MWAA endpoints and quotas in the Amazon Web Services General Reference.

Usage
mwaa(config = list(), credentials = list(), endpoint = NULL, region = NULL)

Arguments
config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
– **session_token**: AWS temporary session token
  
  – **profile**: The name of a profile to use. If not given, then the default profile is used.
  
  – **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

### Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

### Service syntax

```r
svc <- mwaa(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

### Operations
**Examples**

```r
## Not run:
svc <- mwaa()
svc$create_cli_token(
  Foo = 123
)
## End(Not run)
```

---

**neptune**

### Amazon Neptune

Amazon Neptune is a fast, reliable, fully-managed graph database service that makes it easy to build and run applications that work with highly connected datasets. The core of Amazon Neptune is a purpose-built, high-performance graph database engine optimized for storing billions of relationships and querying the graph with milliseconds latency. Amazon Neptune supports popular graph models Property Graph and W3C’s RDF, and their respective query languages Apache TinkerPop Gremlin and SPARQL, allowing you to easily build queries that efficiently navigate highly connected datasets. Neptune powers graph use cases such as recommendation engines, fraud detection, knowledge graphs, drug discovery, and network security.

This interface reference for Amazon Neptune contains documentation for a programming or command line interface you can use to manage Amazon Neptune. Note that Amazon Neptune is asynchronous, which means that some interfaces might require techniques such as polling or callback functions to determine when a command has been applied. In this reference, the parameter descriptions indicate whether a command is applied immediately, on the next instance reboot, or during the maintenance window. The reference structure is as follows, and we list following some related topics from the user guide.
neptune

Usage

neptune(config = list(), credentials = list(), endpoint = NULL, region = NULL)

Arguments

config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

default
  • endpoint: Optional shorthand for complete URL to use for the constructed client.
  • region: Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```r
svc <- neptune(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `add_role_to_db_cluster` Associates an Identity and Access Management (IAM) role with an Neptune DB cluster.
- `add_source_identifier_to_subscription` Adds a source identifier to an existing event notification subscription.
- `add_tags_to_resource` Adds metadata tags to an Amazon Neptune resource.
- `apply_pending_maintenance_action` Applies a pending maintenance action to a resource (for example, to a DB instance).
- `copy_db_cluster_parameter_group` Copies the specified DB cluster parameter group.
- `copy_db_cluster_snapshot` Copies a snapshot of a DB cluster.
- `copy_db_cluster` Creates a new Amazon Neptune DB cluster.
- `create_db_cluster_endpoint` Creates a new custom endpoint and associates it with an Amazon Neptune DB cluster.
- `create_db_cluster_parameter_group` Creates a new DB cluster parameter group.
- `create_db_cluster_snapshot` Creates a snapshot of a DB cluster.
- `create_db_instance` Creates a new DB instance.
- `create_db_parameter_group` Creates a new DB parameter group.
- `create_db_subnet_group` Creates a new DB subnet group.
create_event_subscription
create_global_cluster
delete_db_cluster
delete_db_cluster_endpoint
delete_db_cluster_parameter_group
delete_db_cluster_snapshot
delete_db_instance
delete_db_parameter_group
delete_db_subnet_group
delete_event_subscription
delete_global_cluster
describe_db_cluster_endpoints
describe_db_cluster_parameter_groups
describe_db_cluster_parameters
describe_db_clusters
describe_db_cluster_snapshot_attributes
describe_db_cluster_snapshots
describe_db_engine_versions
describe_db_instances
describe_db_parameter_groups
describe_db_parameters
describe_db_subnet_groups
describe_engine_default_cluster_parameters
describe_engine_default_parameters
describe_event_categories
describe_events
describe_event_subscriptions
describe_global_clusters
describe_orderable_db_instance_options
describe_pending_maintenance_actions
describe_valid_db_instance_modifications
failover_db_cluster
failover_global_cluster
list_tags_for_resource
modify_db_cluster
modify_db_cluster_endpoint
modify_db_cluster_parameter_group
modify_db_cluster_snapshot_attribute
modify_db_instance
modify_db_parameter_group
modify_db_subnet_group
modify_event_subscription
modify_global_cluster
promote_read_replica_db_cluster
reboot_db_instance
remove_from_global_cluster
remove_role_from_db_cluster
remove_source_identifier_from_subscription

Creates an event notification subscription
Creates a Neptune global database spread across multiple Amazon Regions
The DeleteDBCluster action deletes a previously provisioned DB cluster
Deletes a custom endpoint and removes it from an Amazon Neptune DB cluster
Deletes a specified DB cluster parameter group
Deletes a DB cluster snapshot
The DeleteDBInstance action deletes a previously provisioned DB instance
Deletes a specified DBParameterGroup
Deletes a DB subnet group
Deletes an event notification subscription
Deletes a global database
Returns information about endpoints for an Amazon Neptune DB cluster
Returns a list of DBClusterParameterGroup descriptions
Returns the detailed parameter list for a particular DB cluster parameter group
Returns information about provisioned DB clusters, and supports pagination
Returns a list of DB cluster snapshot attribute names and values for a manual DB cluster snapshot
Returns information about DB cluster snapshots
Returns a list of the available DB engines
Returns information about provisioned instances, and supports pagination
Returns a list of DBParameterGroup descriptions
Returns the detailed parameter list for a particular DB parameter group
Returns a list of DBSubnetGroup descriptions
Returns the default engine and system parameter information for the cluster database engine
Returns the default engine and system parameter information for the specified database engine
Displays a list of categories for all event source types, or, if specified, for a specific source type
Returns events related to DB instances, DB security groups, DB snapshots, and DB parameter groups
Lists all the subscription descriptions for a customer account
Returns information about Neptune global database clusters
Returns a list of orderable DB instance options for the specified engine
Returns a list of resources (for example, DB instances) that have at least one pending maintenance action
You can call DescribeValidDBInstanceModifications to learn what modifications you can make to your DB instance
Forces a failover for a DB cluster
Initiates the failover process for a Neptune global database
Lists all tags on an Amazon Neptune resource
Modify a setting for a DB cluster
Modifies the properties of an endpoint in an Amazon Neptune DB cluster
Modifies the parameters of a DB cluster parameter group
Adds an attribute and values to, or removes an attribute and values from, a manual DB cluster snapshot
Modifies settings for a DB instance
Modifies the parameters of a DB parameter group
Modifies an existing DB subnet group
Modifies an existing event notification subscription
Modify a setting for an Amazon Neptune global cluster
Not supported
You might need to reboot your DB instance, usually for maintenance reasons
Detaches a Neptune DB cluster from a Neptune global database
Disassociates an Identity and Access Management (IAM) role from a DB cluster
Removes a source identifier from an existing event notification subscription
**remove_tags_from_resource**
Removes metadata tags from an Amazon Neptune resource

**reset_db_cluster_parameter_group**
Modifies the parameters of a DB cluster parameter group to the default value

**reset_db_parameter_group**
Modifies the parameters of a DB parameter group to the engine/system default value

**restore_db_cluster_from_snapshot**
Creates a new DB cluster from a DB snapshot or DB cluster snapshot

**restore_db_cluster_to_point_in_time**
Restores a DB cluster to an arbitrary point in time

**start_db_cluster**
Starts an Amazon Neptune DB cluster that was stopped using the Amazon console, the Amazon CLI `stop-db-cluster` command, or the StopDBCluster API

**stop_db_cluster**
Stops an Amazon Neptune DB cluster

---

#### Examples

```r
## Not run:
svc <- neptune()
svc$add_role_to_db_cluster(
  Foo = 123
)
## End(Not run)
```

---

**neptunedata**

*Amazon NeptuneData*

---

**Description**

Neptune Data API

The Amazon Neptune data API provides SDK support for more than 40 of Neptune’s data operations, including data loading, query execution, data inquiry, and machine learning. It supports the Gremlin and openCypher query languages, and is available in all SDK languages. It automatically signs API requests and greatly simplifies integrating Neptune into your applications.

**Usage**

```r
neptunedata(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

**Arguments**

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
* secret_access_key: AWS secret access key
* session_token: AWS temporary session token
– profile: The name of a profile to use. If not given, then the default profile is used.
– anonymous: Set anonymous credentials.

• endpoint: The complete URL to use for the constructed client.
• region: The AWS Region used in instantiating the client.
• close_connection: Immediately close all HTTP connections.
• timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter
• creds:
  – access_key_id: AWS access key ID
  – secret_access_key: AWS secret access key
  – session_token: AWS temporary session token
• profile: The name of a profile to use. If not given, then the default profile is used.
• anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

svc <- neptunedata(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
  )
)
region = "string",
close_connection = "logical",
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

cancel_gremlin_query Cancels a Gremlin query
cancel_loader_job Cancels a specified load job
cancel_ml_data_processing_job Cancels a Neptune ML data processing job
cancel_ml_model_training_job Cancels a Neptune ML model training job
cancel_ml_model_transform_job Cancels a specified model transform job
create_ml_endpoint Creates a new Neptune ML inference endpoint that lets you query one specific model that the model-training process constructed
cancel_ml_endpoint Cancels the creation of a Neptune ML inference endpoint
delete_ml_endpoint Cancels the creation of a Neptune ML inference endpoint
delete_propertygraph_statistics Deletes statistics for Gremlin and openCypher (property graph) data
delete_sparql_statistics Deletes SPARQL statistics
delete_ml_model_transform_job Gets information about a specified model transform job
delete_propertygraph_stream Gets a stream for a property graph
delete_propertygraph_summary Gets a graph summary for a property graph
execute_fast_reset The fast reset REST API lets you reset a Neptune graph quickly and easily, removing all of its data
execute_gremlin_explain_query Executes a Gremlin Explain query
execute_gremlin_profile_query Executes a Gremlin Profile query, which runs a specified traversal, collects various metrics about the run, and produces a profile report as output
execute_gremlin_query This command executes a Gremlin query
execute_open_cypher_explain_query Executes an openCypher explain request
execute_open_cypher_query Executes an openCypher query
get_engine_status Retrieves the status of the graph database on the host
get_gremlin_query_status Gets the status of a specified Gremlin query
get_loader_job_status Gets status information about a specified load job
get_ml_data_processing_job Retrieves information about a specified data processing job
get_ml_endpoint Retrieves details about an inference endpoint
get_ml_model_training_job Retrieves information about a Neptune ML model training job
get_ml_model_transform_job Retrieves information about a specified model transform job
get_open_cypher_query_status Retrieves the status of a specified openCypher query
get_propertygraph_statistics Gets property graph statistics (Gremlin and openCypher)
get_propertygraph_stream Gets a stream for a property graph
get_propertygraph_summary Gets a graph summary for a property graph
### Examples

```r
## Not run:
svc <- neptunedata()
svc$cancel_gremlin_query(
  Foo = 123
)
## End(Not run)
```

---

**networkfirewall**  
**AWS Network Firewall**

### Description

This is the API Reference for Network Firewall. This guide is for developers who need detailed information about the Network Firewall API actions, data types, and errors.

- The REST API requires you to handle connection details, such as calculating signatures, handling request retries, and error handling. For general information about using the Amazon Web Services REST APIs, see [Amazon Web Services APIs](https://aws.amazon.com/api).

To access Network Firewall using the REST API endpoint: `https://network-firewall.<region>.amazonaws.com`

- Alternatively, you can use one of the Amazon Web Services SDKs to access an API that’s tailored to the programming language or platform that you’re using. For more information, see [Amazon Web Services SDKs](https://aws.amazon.com/sdk).

- For descriptions of Network Firewall features, including and step-by-step instructions on how to use them through the Network Firewall console, see the [Network Firewall Developer Guide](https://aws.amazon.com/network-firewall/developer-guide).
Network Firewall is a stateful, managed, network firewall and intrusion detection and prevention service for Amazon Virtual Private Cloud (Amazon VPC). With Network Firewall, you can filter traffic at the perimeter of your VPC. This includes filtering traffic going to and coming from an internet gateway, NAT gateway, or over VPN or Direct Connect. Network Firewall uses rules that are compatible with Suricata, a free, open source network analysis and threat detection engine. Network Firewall supports Suricata version 6.0.9. For information about Suricata, see the Suricata website.

You can use Network Firewall to monitor and protect your VPC traffic in a number of ways. The following are just a few examples:

- Allow domains or IP addresses for known Amazon Web Services service endpoints, such as Amazon S3, and block all other forms of traffic.
- Use custom lists of known bad domains to limit the types of domain names that your applications can access.
- Perform deep packet inspection on traffic entering or leaving your VPC.
- Use stateful protocol detection to filter protocols like HTTPS, regardless of the port used.

To enable Network Firewall for your VPCs, you perform steps in both Amazon VPC and in Network Firewall. For information about using Amazon VPC, see Amazon VPC User Guide.

To start using Network Firewall, do the following:

1. (Optional) If you don’t already have a VPC that you want to protect, create it in Amazon VPC.
2. In Amazon VPC, in each Availability Zone where you want to have a firewall endpoint, create a subnet for the sole use of Network Firewall.
3. In Network Firewall, create stateless and stateful rule groups, to define the components of the network traffic filtering behavior that you want your firewall to have.
4. In Network Firewall, create a firewall policy that uses your rule groups and specifies additional default traffic filtering behavior.
5. In Network Firewall, create a firewall and specify your new firewall policy and VPC subnets. Network Firewall creates a firewall endpoint in each subnet that you specify, with the behavior that’s defined in the firewall policy.
6. In Amazon VPC, use ingress routing enhancements to route traffic through the new firewall endpoints.

Usage

```
networkfirewall(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:


- **creds:**
  - *access_key_id*: AWS access key ID
  - *secret_access_key*: AWS secret access key
  - *session_token*: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.

- **anonymous**: Set anonymous credentials.

  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

**credentials**

Optional credentials shorthand for the config parameter

- **creds**:
  - *access_key_id*: AWS access key ID
  - *secret_access_key*: AWS secret access key
  - *session_token*: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.

- **anonymous**: Set anonymous credentials.

**endpoint**

Optional shorthand for complete URL to use for the constructed client.

**region**

Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- networkfirewall(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = false
    ),
    region = "string",
    endpoint = "string"
  )
)```
networkfirewall

Anonymous = "logical",
),
endpoint = "string",
region = "string",
close_connection = "logical",
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
creds = list(
  access_key_id = "string",
  secret_access_key = "string",
  session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

associate_firewall_policy
associate_subnets
create_firewall
create_firewall_policy
create_rule_group
create_tls_inspection_configuration
delete_firewall
delete_firewall_policy
delete_resource_policy
delete_rule_group
delete_tls_inspection_configuration
describe_firewall
describe_firewall_policy
describe_logging_configuration
describe_resource_policy
describe_rule_group
describe_rule_group_metadata
describe_tls_inspection_configuration
disassociate_subnets
list_firewall_policies
list_firewalls
list_rule_groups
list_tags_for_resource
list_tls_inspection_configurations

Associates a FirewallPolicy to a Firewall
Associates the specified subnets in the Amazon VPC to the firewall
Creates an Network Firewall Firewall and accompanying FirewallStatus for a VPC
Creates the firewall policy for the firewall according to the specifications
Creates an Network Firewall TLS inspection configuration
Deletes the specified Firewall and its FirewallStatus
Deletes a resource policy that you created in a PutResourcePolicy request
Deletes the specified RuleGroup
Deletes the specified TLSInspectionConfiguration
Returns the data objects for the specified firewall policy
Returns the data objects for the specified firewall
Returns the logging configuration for the specified firewall
Returns a policy that you created in a PutResourcePolicy request
Returns the data objects for the specified rule group
High-level information about a rule group, returned by operations like create and describe
Returns the data objects for the specified TLS inspection configuration
Removes the specified subnet associations from the firewall
Retrieves the metadata for the firewall policies that you have defined
Retrieves the metadata for the firewall that you defined
Retrieves the tags associated with the specified resource
Retrieves the metadata for the TLS inspection configurations that you have defined
Examples

```r
## Not run:
svc <- networkfirewall()
svc$associate_firewall_policy(
  Foo = 123
)
## End(Not run)
```

---

### Description

Amazon Web Services enables you to centrally manage your Amazon Web Services Cloud WAN core network and your Transit Gateway network across Amazon Web Services accounts, Regions, and on-premises locations.

### Usage

```r
networkmanager(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```
Arguments

config Optional configuration of credentials, endpoint, and/or region.
   • credentials:
     – creds:
       * access_key_id: AWS access key ID
       * secret_access_key: AWS secret access key
       * session_token: AWS temporary session token
     – profile: The name of a profile to use. If not given, then the default profile is used.
     – anonymous: Set anonymous credentials.
   • endpoint: The complete URL to use for the constructed client.
   • region: The AWS Region used in instantiating the client.
   • close_connection: Immediately close all HTTP connections.
   • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
   • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
   • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html

credentials Optional credentials shorthand for the config parameter
   • creds:
     – access_key_id: AWS access key ID
     – secret_access_key: AWS secret access key
     – session_token: AWS temporary session token
   • profile: The name of a profile to use. If not given, then the default profile is used.
   • anonymous: Set anonymous credentials.

documentmanager

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

svc <- networkmanager(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
```
networkmanager

```python
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
endpoint = "string",
region = "string",
)
```

**Operations**

- **accept_attachment**
  Accepts a core network attachment request
- **associate_connect_peer**
  Associates a core network Connect peer with a device and optionally, with a link
- **associate_customer_gateway**
  Associates a customer gateway with a device and optionally, with a link
- **associate_link**
  Associates a link to a device
- **associate_transit_gateway_connect_peer**
  Associates a transit gateway Connect peer with a device, and optionally, with a link
- **create_connect_attachment**
  Creates a core network Connect attachment from a specified core network attachment
- **create_connection**
  Creates a connection between two devices
- **create_connect_peer**
  Creates a core network Connect peer for a specified core network attachment between a core network and an appliance
- **create_core_network**
  Creates a new device in a global network
- **create_global_network**
  Creates a new, empty global network
- **create_link**
  Creates a new link for a specified site
- **create_site**
  Creates a new site in a global network
- **create_site_to_site_vpn_attachment**
  Creates an Amazon Web Services site-to-site VPN attachment on an edge location of a core network
- **create_transit_gateway_peering**
  Creates a transit gateway peering connection
- **create_transit_gateway_route_table_attachment**
  Creates a transit gateway route table attachment
- **create_vpc_attachment**
  Creates a VPC attachment on an edge location of a core network
- **delete_attachment**
  Deletes an attachment
- **delete_connection**
  Deletes the specified connection in your global network
- **delete_connect_peer**
  Deletes a Connect peer
delete_core_network
delete_core_network_policy_version
delete_device
delete_global_network
delete_link
delete_peering
delete_resource_policy
delete_site
deregister_transit_gateway
describe_global_networks
disassociate_connect_peer
disassociate_customer_gateway
disassociate_link
disassociate_transit_gateway_connect_peer
execute_core_network_change_set
get_connect_attachment
get_connections
get_connect_peer
get_connect_peer_associations
get_core_network
get_core_network_change_events
get_core_network_change_set
get_core_network_policy
get_customer_gateway_associations
get_devices
get_link_associations
get_links
get_network_resource_counts
get_network_resource_relationships
get_network_resources
get_network_routes
get_network_telemetry
get_resource_policy
get_route_analysis
get_sites
get_site_to_site_vpn_attachment
get_transit_gateway_connect_peer_associations
get_transit_gateway_peering
get_transit_gatewayRegistrations
get_transit_gateway_route_table_attachment
get_vpc_attachment
list_attachments
list_connect_peers
list_core_network_policy_versions
list_core_networks
list_organization_service_access_status
list_peerings
list_tags_for_resource

Delete a core network along with all core network policies
Delete a policy version from a core network
Delete an existing device
Delete an existing global network
Delete an existing link
Delete an existing peering connection
Delete a resource policy for the specified resource
Delete an existing site
Deregisters a transit gateway from your global network
Describes one or more global networks
Disassociates a core network Connect peer from a device and a link
Disassociates a customer gateway from a device and a link
Disassociates an existing device from a link
Disassociates a transit gateway Connect peer from a device and link
Executes a change set on your core network
Returns information about a core network Connect attachment
Gets information about one or more of your connections in a global network
Returns information about a core network Connect peer
Returns information about a core network Connect peer associations
Returns information about the LIVE policy for a core network
Returns information about a core network change event
Returns a change set between the LIVE core network policy and a submitted
Returns details about a core network policy
Gets the association information for customer gateways that are associated with devices and links
Gets information about one or more of your devices in a global network
Gets the link associations for a device or a link
Gets information about one or more links in a specified global network
Gets the count of network resources, by resource type, for the specified global network
Gets the network resource relationships for the specified global network
Describes the network resources for the specified global network
Gets the network routes of the specified global network
Gets the network telemetry of the specified global network
Returns information about a resource policy
Gets information about the specified route analysis
Gets information about one or more of your sites in a global network
Returns information about a site-to-site VPN attachment
Gets information about one or more of your transit gateway Connect peers
Returns information about a transit gateway peer
Gets information about the transit gateway registrations in a specified global network
Returns information about a transit gateway route table attachment
Returns information about a VPC attachment
Returns a list of core network attachments
Returns a list of core network Connect peers
Returns a list of core network policy versions
Returns a list of owned and shared core networks
Gets the status of the Service Linked Role (SLR) deployment for the account
Lists the peerings for a core network
Lists the tags for a specified resource
nimblestudio

put_core_network_policy
put_resource_policy
register_transit_gateway
reject_attachment
restore_core_network_policy_version
start_organization_service_access_update
start_route_analysis
tag_resource
untag_resource
update_connection
update_core_network
update_device
update_global_network
update_link
update_network_resource_metadata
update_site
update_vpc_attachment

Creates a new, immutable version of a core network policy
Creates or updates a resource policy
Registers a transit gateway in your global network
Rejects a core network attachment request
Restores a previous policy version as a new, immutable version of a core network policy
Enables the Network Manager service for an Amazon Web Services Organization
Starts analyzing the routing path between the specified source and destination
Tags a specified resource
Removes tags from a specified resource
Updates the information for an existing connection
Updates the description of a core network
Updates the details for an existing device
Updates an existing global network
Updates the details for an existing link
Updates the resource metadata for the specified global network
Updates the information for an existing site
Updates a VPC attachment

Examples

```r
## Not run:
svc <- networkmanager()
svc$accept_attachment(
  Foo = 123
)

## End(Not run)
```

---

nimblestudio  AmazonNimbleStudio

Description

Welcome to the Amazon Nimble Studio API reference. This API reference provides methods, schema, resources, parameters, and more to help you get the most out of Nimble Studio.

Nimble Studio is a virtual studio that empowers visual effects, animation, and interactive content teams to create content securely within a scalable, private cloud service.

Usage

```r
nimblestudio(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```
Arguments

config Optional configuration of credentials, endpoint, and/or region.
   • credentials:
      – creds:
         * access_key_id: AWS access key ID
         * secret_access_key: AWS secret access key
         * session_token: AWS temporary session token
      – profile: The name of a profile to use. If not given, then the default profile is used.
      – anonymous: Set anonymous credentials.
   • endpoint: The complete URL to use for the constructed client.
   • region: The AWS Region used in instantiating the client.
   • close_connection: Immediately close all HTTP connections.
   • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
   • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
   • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter
   • creds:
      – access_key_id: AWS access key ID
      – secret_access_key: AWS secret access key
      – session_token: AWS temporary session token
   • profile: The name of a profile to use. If not given, then the default profile is used.
   • anonymous: Set anonymous credentials.

edpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

svc <- nimblestudio(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
      )
    )
  )
)
secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
),
    credentials = list(
        creds = list(
            access_key_id = "string",
            secret_access_key = "string",
            session_token = "string"
        ),
        profile = "string",
        anonymous = "logical"
    ),
    endpoint = "string",
    region = "string"
)

Operations

accept_eulas       Accept EULAs
create_launch_profile Create a launch profile
create_streaming_image Creates a streaming image resource in a studio
create_streaming_session Creates a streaming session in a studio
create_streaming_session_stream Creates a streaming session stream for a streaming session
create_studio Create a new studio
create_studio_component Creates a studio component resource
delete_launch_profile Permanently delete a launch profile
delete_launch_profile_member Delete a user from launch profile membership
delete_streaming_image Delete streaming image
delete_streaming_session Deletes streaming session resource
delete_studio Delete a studio resource
delete_studio_component Deletes a studio component resource
delete_studio_member Delete a user from studio membership
get_eula Get EULA
get_launch_profile Get a launch profile
get_launch_profile_details Launch profile details include the launch profile resource and summary information of resources that are used by, or available to, the launch profile
get_launch_profile_initialization Get a launch profile initialization
get_launch_profile_member Get a user persona in launch profile membership
get_streaming_image Get streaming image
<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>get_streaming_session</td>
<td>Gets StreamingSession resource</td>
</tr>
<tr>
<td>get_streaming_session_backup</td>
<td>Gets StreamingSessionBackup resource</td>
</tr>
<tr>
<td>get_streaming_session_stream</td>
<td>Gets a StreamingSessionStream for a streaming session</td>
</tr>
<tr>
<td>get_studio</td>
<td>Get a studio resource</td>
</tr>
<tr>
<td>get_studio_component</td>
<td>Gets a studio component resource</td>
</tr>
<tr>
<td>get_studio_member</td>
<td>Get a user’s membership in a studio</td>
</tr>
<tr>
<td>list_eula_acceptances</td>
<td>List EULA acceptances</td>
</tr>
<tr>
<td>list_eulas</td>
<td>List EULAs</td>
</tr>
<tr>
<td>list_launch_profile_members</td>
<td>Get all users in a given launch profile membership</td>
</tr>
<tr>
<td>list_launch_profiles</td>
<td>List all the launch profiles a studio</td>
</tr>
<tr>
<td>list_streaming_images</td>
<td>List the streaming image resources available to this studio</td>
</tr>
<tr>
<td>list_streaming_session_backups</td>
<td>Lists the backups of a streaming session in a studio</td>
</tr>
<tr>
<td>list_streaming_sessions</td>
<td>Lists the streaming sessions in a studio</td>
</tr>
<tr>
<td>list_studio_components</td>
<td>Lists the StudioComponents in a studio</td>
</tr>
<tr>
<td>list_studio_members</td>
<td>Get all users in a given studio membership</td>
</tr>
<tr>
<td>list_studios</td>
<td>List studios in your Amazon Web Services accounts in the requested Amazon Web Service region</td>
</tr>
<tr>
<td>list_tags_for_resource</td>
<td>Gets the tags for a resource, given its Amazon Resource Names (ARN)</td>
</tr>
<tr>
<td>put_launch_profile_members</td>
<td>Add/update users with given persona to launch profile membership</td>
</tr>
<tr>
<td>put_studio_members</td>
<td>Add/update users with given persona to studio membership</td>
</tr>
<tr>
<td>start_streaming_session</td>
<td>Transitions sessions from the STOPPED state into the READY state</td>
</tr>
<tr>
<td>start_studio_sso_configuration_repair</td>
<td>Repairs the IAM Identity Center configuration for a given studio</td>
</tr>
<tr>
<td>stop_streaming_session</td>
<td>Transitions sessions from the READY state into the STOPPED state</td>
</tr>
<tr>
<td>tag_resource</td>
<td>Creates tags for a resource, given its ARN</td>
</tr>
<tr>
<td>untag_resource</td>
<td>Deletes the tags for a resource</td>
</tr>
<tr>
<td>update_launch_profile</td>
<td>Update a launch profile</td>
</tr>
<tr>
<td>update_launch_profile_member</td>
<td>Update a user persona in launch profile membership</td>
</tr>
<tr>
<td>update_streaming_image</td>
<td>Update streaming image</td>
</tr>
<tr>
<td>update_studio</td>
<td>Update a Studio resource</td>
</tr>
<tr>
<td>update_studio_component</td>
<td>Updates a studio component resource</td>
</tr>
</tbody>
</table>

**Examples**

```r
## Not run:
svc <- nimblestudio()
svc$accept_eulas(
    Foo = 123
)

## End(Not run)
```

---

**omics**  
*Amazon Omics*
**Description**

This is the *AWS HealthOmics API Reference*. For an introduction to the service, see [What is AWS HealthOmics?](https://aws.amazon.com) in the [AWS HealthOmics User Guide](https://aws.amazon.com).

**Usage**

```r
omics(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

**Arguments**

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**: 
    - **creds**: 
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**: 
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

- **endpoint**: Optional shorthand for complete URL to use for the constructed client.

- **region**: Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```r
svc <- omics(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `abort_multipart_read_set_upload`: Stops a multipart upload
- `accept_share`: Accept a resource share request
- `batch_delete_read_set`: Deletes one or more read sets
- `cancel_annotation_import_job`: Cancels an annotation import job
- `cancel_run`: Cancels a run
- `cancel_variant_import_job`: Cancels a variant import job
- `complete_multipart_read_set_upload`: Concludes a multipart upload once you have uploaded all the components
- `create_annotation_store`: Creates an annotation store
- `create_annotation_store_version`: Creates a new version of an annotation store
- `create_multipart_read_set_upload`: Begins a multipart read set upload
- `create_reference_store`: Creates a reference store
- `create_run_group`: Creates a run group
- `create_sequence_store`: Creates a sequence store
- `create_share`: Creates a cross-account shared resource
create_variant_store
create_workflow
delete_annotation_store
delete_annotation_store_versions
delete_reference
delete_reference_store
delete_run
delete_run_group
delete_sequence_store
delete_share
delete_variant_store
delete_workflow
get_annotation_import_job
get_annotation_store
get_annotation_store_version
get_read_set
get_read_set_activation_job
get_read_set_export_job
get_read_set_import_job
get_read_set_metadata
get_reference
get_reference_import_job
get_reference_metadata
get_reference_store
get_run
get_run_group
get_sequence_store
get_share
get_variant_import_job
get_variant_store
get_workflow
list_annotation_import_jobs
list_annotation_stores
list_annotation_store_versions
list_multipart_read_set_uploads
list_read_set_activation_jobs
list_read_set_export_jobs
list_read_set_import_jobs
list_read_sets
list_read_set_upload_parts
list_reference_import_jobs
list_references
list_reference_stores
list_run_groups
list_runs
list_run_tasks
list_sequence_stores

create_variant_store
create_workflow
delete_annotation_store
delete_annotation_store_versions
delete_reference
delete_reference_store
delete_run
delete_run_group
delete_sequence_store
delete_share
delete_variant_store
delete_workflow
get_annotation_import_job
get_annotation_store
get_annotation_store_version
get_read_set
get_read_set_activation_job
get_read_set_export_job
get_read_set_import_job
get_read_set_metadata
get_reference
get_reference_import_job
get_reference_metadata
get_reference_store
get_run
get_run_group
get_sequence_store
get_share
get_variant_import_job
get_variant_store
get_workflow
list_annotation_import_jobs
list_annotation_stores
list_annotation_store_versions
list_multipart_read_set_uploads
list_read_set_activation_jobs
list_read_set_export_jobs
list_read_set_import_jobs
list_read_sets
list_read_set_upload_parts
list_reference_import_jobs
list_references
list_reference_stores
list_run_groups
list_runs
list_run_tasks
list_sequence_stores

Creates a variant store
Creates a workflow
Deletes an annotation store
Deletes one or multiple versions of an annotation store
Deletes a genome reference
Deletes a genome reference store
Deletes a workflow run
Deletes a workflow run group
Deletes a sequence store
Deletes a resource share
Deletes a variant store
Deletes a workflow
Gets information about an annotation import job
Gets information about an annotation store
Retrieves the metadata for an annotation store version
Gets a file from a read set
Gets information about a read set activation job
Gets information about a read set export job
Gets information about a read set import job
Gets details about a read set
Gets a reference file
Gets information about a reference import job
Gets information about a genome reference’s metadata
Gets information about a reference store
Gets information about a workflow run
Gets information about a workflow run group
Gets information about a workflow run task
Gets information about a sequence store
Retrieves the metadata for the specified resource share
Gets information about a variant import job
Gets information about a variant store
Gets information about a workflow
Retrieves a list of annotation import jobs
Retrieves a list of annotation stores
Lists the versions of an annotation store
Lists multipart read set uploads and for in progress uploads
Retrieves a list of read set activation jobs
Retrieves a list of read set export jobs
Retrieves a list of read set import jobs
Retrieves a list of read sets
This operation will list all parts in a requested multipart upload for a sequence store
Retrieves a list of reference import jobs
Retrieves a list of references
Retrieves a list of reference stores
Retrieves a list of run groups
Retrieves a list of runs
Retrieves a list of tasks for a run
Retrieves a list of sequence stores
<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>list_shares</td>
<td>Retrieves the resource shares associated with an account</td>
</tr>
<tr>
<td>list_tags_for_resource</td>
<td>Retrieves a list of tags for a resource</td>
</tr>
<tr>
<td>list_variant_import_jobs</td>
<td>Retrieves a list of variant import jobs</td>
</tr>
<tr>
<td>list_variant_stores</td>
<td>Retrieves a list of variant stores</td>
</tr>
<tr>
<td>list_workflows</td>
<td>Retrieves a list of workflows</td>
</tr>
<tr>
<td>start_annotation_import_job</td>
<td>Starts an annotation import job</td>
</tr>
<tr>
<td>start_read_set_activation_job</td>
<td>Activates an archived read set</td>
</tr>
<tr>
<td>start_read_set_export_job</td>
<td>Exports a read set to Amazon S3</td>
</tr>
<tr>
<td>start_read_set_import_job</td>
<td>Starts a read set import job</td>
</tr>
<tr>
<td>start_reference_import_job</td>
<td>Starts a reference import job</td>
</tr>
<tr>
<td>start_run</td>
<td>Starts a workflow run</td>
</tr>
<tr>
<td>start_variant_import_job</td>
<td>Starts a variant import job</td>
</tr>
<tr>
<td>tag_resource</td>
<td>Tags a resource</td>
</tr>
<tr>
<td>untag_resource</td>
<td>Removes tags from a resource</td>
</tr>
<tr>
<td>update_annotation_store</td>
<td>Updates an annotation store</td>
</tr>
<tr>
<td>update_annotation_store_version</td>
<td>Updates the description of an annotation store version</td>
</tr>
<tr>
<td>update_run_group</td>
<td>Updates a run group</td>
</tr>
<tr>
<td>update_variant_store</td>
<td>Updates a variant store</td>
</tr>
<tr>
<td>update_workflow</td>
<td>Updates a workflow</td>
</tr>
<tr>
<td>upload_read_set_part</td>
<td>This operation uploads a specific part of a read set</td>
</tr>
</tbody>
</table>

**Examples**

```r
## Not run:
svc <- omics()
svc$abort_multipart_read_set_upload(
  Foo = 123
)

## End(Not run)
```

---

**Amazon OpenSearch Ingestion**

**Description**

Use the Amazon OpenSearch Ingestion API to create and manage ingestion pipelines. OpenSearch Ingestion is a fully managed data collector that delivers real-time log and trace data to OpenSearch Service domains. For more information, see Getting data into your cluster using OpenSearch Ingestion.
opensearchingestion

Usage

```python
opensearchingestion(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **creds**:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - `profile`: The name of a profile to use. If not given, then the default profile is used.
  - `anonymous`: Set anonymous credentials.
- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - `profile`: The name of a profile to use. If not given, then the default profile is used.
  - `anonymous`: Set anonymous credentials.
- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```
svc <- opensearchingestion(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- create_pipeline: Creates an OpenSearch Ingestion pipeline
- delete_pipeline: Deletes an OpenSearch Ingestion pipeline
- get_pipeline: Retrieves information about an OpenSearch Ingestion pipeline
- get_pipeline_blueprint: Retrieves information about a specific blueprint for OpenSearch Ingestion
- get_pipeline_change_progress: Returns progress information for the current change happening on an OpenSearch Ingestion pipeline
- list_pipeline_blueprints: Retrieves a list of all available blueprints for Data Prepper
- list_pipelines: Lists all OpenSearch Ingestion pipelines in the current Amazon Web Services account and Region
- list_tags_for_resource: Lists all resource tags associated with an OpenSearch Ingestion pipeline
- start_pipeline: Starts an OpenSearch Ingestion pipeline
- stop_pipeline: Stops an OpenSearch Ingestion pipeline
- tag_resource: Tags an OpenSearch Ingestion pipeline
- untag_resource: Removes one or more tags from an OpenSearch Ingestion pipeline
- update_pipeline: Updates an OpenSearch Ingestion pipeline
- validate_pipeline: Checks whether an OpenSearch Ingestion pipeline configuration is valid prior to creation
opensearchservice

Examples

```r
## Not run:
svc <- opensearchingestion()
svc$create_pipeline(
  Foo = 123
)
## End(Not run)
```

---

opensearchservice  Amazon OpenSearch Service

Description

Use the Amazon OpenSearch Service configuration API to create, configure, and manage OpenSearch Service domains. The endpoint for configuration service requests is Region specific: es.region.amazonaws.com. For example, es.us-east-1.amazonaws.com. For a current list of supported Regions and endpoints, see Amazon Web Services service endpoints.

Usage

```r
opensearchservice(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- `config`: Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
    - `creds`:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
    - `anonymous`: Set anonymous credentials.
  - `endpoint`: The complete URL to use for the constructed client.
  - `region`: The AWS Region used in instantiating the client.
  - `close_connection`: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

• **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

**credentials**

Optional credentials shorthand for the config parameter

• **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

**endpoint**

Optional shorthand for complete URL to use for the constructed client.

**region**

Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- opensearchservice(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  )
)
```
secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

accept_inbound_connection Allows the destination Amazon OpenSearch Service domain owner to accept an inbound cross-cluster search connection request
add_data_source Creates a new direct-query data source to the specified domain
add_tags Attaches tags to an existing Amazon OpenSearch Service domain
associate_package Associates a package with an Amazon OpenSearch Service domain
authorize_vpc_endpoint_access Provides access to an Amazon OpenSearch Service domain through the use of an interface VPC endpoint
cancel_domain_config_change Cancels a pending configuration change on an Amazon OpenSearch Service domain
cancel_service_software_update Cancels a scheduled service software update for an Amazon OpenSearch Service domain
create_domain Creates an Amazon OpenSearch Service domain
create_outbound_connection Creates a new cross-cluster search connection from a source Amazon OpenSearch Service domain to a destination Amazon OpenSearch Service domain
create_package Creates a package for use with Amazon OpenSearch Service domains
create_vpc_endpoint Creates an Amazon OpenSearch Service-managed VPC endpoint
delete_data_source Deletes a direct-query data source
delete_domain Deletes an Amazon OpenSearch Service domain and all of its data
delete_inbound_connection Allows the destination Amazon OpenSearch Service domain owner to delete an existing inbound cross-cluster search connection
delete_outbound_connection Allows the source Amazon OpenSearch Service domain owner to delete an existing outbound cross-cluster search connection
delete_package Deletes an Amazon OpenSearch Service package
delete_vpc_endpoint Deletes an Amazon OpenSearch Service-managed interface VPC endpoint
describe_domain Describes the domain configuration for the specified Amazon OpenSearch Service domain, including the domain ID, domain service endpoint, and domain ARN
describe_domain_auto_tunes Returns the list of optimizations that Auto-Tune has made to an Amazon OpenSearch Service domain
describe_domain_change_progress Returns information about the current blue/green deployment happening on an Amazon OpenSearch Service domain
describe_domain_config Returns the configuration of an Amazon OpenSearch Service domain
describe_domain_health Returns information about domain and node health, the standby Availability Zone, number of nodes per Availability Zone, and shard count per node
describe_domain_nodes Returns information about domain and nodes, including data nodes, master nodes, ultrawarm nodes, Availability Zone(s), standby nodes, node configurations, and node states
describe_domains Returns domain configuration information about the specified Amazon OpenSearch Service domains
describe_dry_run_progress Describes the progress of a pre-update dry run analysis on an Amazon OpenSearch Service domain
describe_inbound_connections Lists the inbound cross-cluster search connections for a destination (remote) Amazon OpenSearch Service domain
describe_instance_type_limits Describes the instance count, storage, and master node limits for a given OpenSearch Service version and instance type
describe_packages Describes all packages available to OpenSearch Service
describeReserved_instance_offerings Describes the available Amazon OpenSearch Service Reserved Instance offerings for a given OpenSearch Service version
describe_reserved_instances Describes one or more Amazon OpenSearch Service-managed VPC endpoints
dissociate_package Removes a package from the specified Amazon OpenSearch Service domain
describe_compatible_versions Returns a map of OpenSearch or Elasticsearch versions and the versions you can upgrade to
describe_data_source Retrieves information about a direct query data source
describe_domain_maintenance_status The status of the maintenance action
get_package_version_history
get_upgrade_history
get_upgrade_status
list_data_sources
list_domain_maintenances
list_domain_names
list_domains_for_package
list_instance_type_details
list_packages_for_domain
list_scheduled_actions
list_tags
list_versions
list_vpc_endpoint_access
list_vpc_endpoints
list_vpc_endpoints_for_domain
purchase_reserved_instance_offering
reject_inbound_connection
remove_tags
revoke_vpc_endpoint_access
start_domain_maintenance
start_service_software_update
update_data_source
update_domain_config
update_package
update_scheduled_action
update_vpc_endpoint
upgrade_domain

Returns a list of Amazon OpenSearch Service package versions, along with their creation time, commit message, and plugin properties (if the package is a zip plugin package)
Retrieves the complete history of the last 10 upgrades performed on an Amazon OpenSearch Service domain
Returns the most recent status of the last upgrade or upgrade eligibility check performed
Lists direct-query data sources for a specific domain
A list of maintenance actions for the domain
Returns the names of all Amazon OpenSearch Service domains owned by the current user
Lists all Amazon OpenSearch Service domains associated with a given package
Lists all instance types and available features for a given OpenSearch or Elasticsearch version
Lists all packages associated with an Amazon OpenSearch Service domain
Retrieves a list of configuration changes that are scheduled for a domain
Returns all resource tags for an Amazon OpenSearch Service domain
Lists all versions of OpenSearch and Elasticsearch that Amazon OpenSearch Service supports
Retrieves information about each Amazon Web Services principal that is allowed to access an Amazon OpenSearch Service domain
Retrieves all Amazon OpenSearch Service-managed VPC endpoints associated with a domain
Allows you to purchase Amazon OpenSearch Service Reserved Instances
Allows the remote Amazon OpenSearch Service domain owner to reject an inbound cross-cluster connection request
Removes the specified set of tags from an Amazon OpenSearch Service domain
Revokes access to an Amazon OpenSearch Service domain that was provided through an interface VPC endpoint
Starts the node maintenance process on the data node
Schedules a service software update for an Amazon OpenSearch Service domain
Updates a direct-query data source
Modifies the cluster configuration of the specified Amazon OpenSearch Service domain
Updates a package for use with Amazon OpenSearch Service domains
Reschedules a planned domain configuration change for a later time
Modifies an Amazon OpenSearch Service-managed interface VPC endpoint
Allows you to either upgrade your Amazon OpenSearch Service domain or perform an upgrade eligibility check.

Examples

```r
## Not run:
svc <- opensearchservice()
svc$accept_inbound_connection(
  Foo = 123
)

## End(Not run)
```

---

Go to top of page

opensearchserviceserverless

OpenSearch Service Serverless
Description

Use the Amazon OpenSearch Serverless API to create, configure, and manage OpenSearch Serverless collections and security policies.

OpenSearch Serverless is an on-demand, pre-provisioned serverless configuration for Amazon OpenSearch Service. OpenSearch Serverless removes the operational complexities of provisioning, configuring, and tuning your OpenSearch clusters. It enables you to easily search and analyze petabytes of data without having to worry about the underlying infrastructure and data management.

To learn more about OpenSearch Serverless, see What is Amazon OpenSearch Serverless?

Usage

```python
opensearchserviceserverless(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

Arguments

- **config**
  - Optional configuration of credentials, endpoint, and/or region.
  - **credentials**: A dictionary containing:
    - `creds`
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
    - `anonymous`: Set anonymous credentials.
  - `endpoint`: The complete URL to use for the constructed client.
  - `region`: The AWS Region used in instantiating the client.
  - `close_connection`: Immediately close all HTTP connections.
  - `timeout`: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - `s3_force_path_style`: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
  - `sts_regional_endpoint`: Set sts regional endpoint resolver to regional or legacy `https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html`

- **credentials**
  - Optional credentials shorthand for the config parameter
  - **creds**: A dictionary containing:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
• **profile:** The name of a profile to use. If not given, then the default profile is used.
• **anonymous:** Set anonymous credentials.

```r
event endpoint = Optional shorthand for complete URL to use for the constructed client.
region = Optional shorthand for AWS Region used in instantiating the client.
```

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
csvc <- opensearchserviceserverless(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

**Operations**

- **batch_get_collection**
  Returns attributes for one or more collections, including the collection endpoint and the
batch_get_effective_lifecycle_policy
batch_get_lifecycle_policy
batch_get_vpc_endpoint
create_access_policy
create_collection
create_lifecycle_policy
create_security_config
create_security_policy
create_vpc_endpoint
delete_access_policy
delete_collection
delete_lifecycle_policy
delete_security_config
delete_security_policy
delete_vpc_endpoint
get_access_policy
get_account_settings
get_policies_stats
get_security_config
get_security_policy
list_access_policies
list_collections
list_lifecycle_policies
list_security_configs
list_security_policies
list_tags_for_resource
list_vpc_endpoints
tag_resource
untag_resource
update_access_policy
update_account_settings
update_collection
update_lifecycle_policy
update_security_config
update_security_policy
update_vpc_endpoint

Returns a list of successful and failed retrievals for the OpenSearch Serverless indexes
Returns one or more configured OpenSearch Serverless lifecycle policies
Returns attributes for one or more VPC endpoints associated with the current account
Creates a data access policy for OpenSearch Serverless
Creates a new OpenSearch Serverless collection
Creates a lifecycle policy to be applied to OpenSearch Serverless indexes
Specifies a security configuration for OpenSearch Serverless
Creates a security policy to be used by one or more OpenSearch Serverless collections
Creates an OpenSearch Serverless-managed interface VPC endpoint
Deletes an OpenSearch Serverless access policy
Deletes an OpenSearch Serverless collection
Deletes an OpenSearch Serverless lifecycle policy
Deletes a security configuration for OpenSearch Serverless
Deletes an OpenSearch Serverless security policy
Deletes an OpenSearch Serverless-managed interface endpoint
Returns an OpenSearch Serverless access policy
Returns account-level settings related to OpenSearch Serverless
Returns statistical information about your OpenSearch Serverless access policies, security configurations, and security policies
Returns information about an OpenSearch Serverless security configuration
Returns information about a configured OpenSearch Serverless security policy
Returns information about a list of OpenSearch Serverless access policies
Lists all OpenSearch Serverless collections
Returns a list of OpenSearch Serverless lifecycle policies
Returns information about configured OpenSearch Serverless security configurations
Returns information about configured OpenSearch Serverless security policies
Returns the tags for an OpenSearch Serverless resource
Returns the OpenSearch Serverless-managed interface VPC endpoints associated with the current account
Associates tags with an OpenSearch Serverless resource
Removes a tag or set of tags from an OpenSearch Serverless resource
Updates an OpenSearch Serverless access policy
Updates the OpenSearch Serverless settings for the current Amazon Web Services account
Updates an OpenSearch Serverless collection
Updates an OpenSearch Serverless access policy
Updates a security configuration for OpenSearch Serverless
Updates an OpenSearch Serverless security policy
Updates an OpenSearch Serverless-managed interface endpoint

Examples

## Not run:
svc <- opensearchserviceserverless()
svc$batch_get_collection(
  Foo = 123
)

## End(Not run)
Description

Welcome to the AWS OpsWorks Stacks API Reference. This guide provides descriptions, syntax, and usage examples for AWS OpsWorks Stacks actions and data types, including common parameters and error codes.

AWS OpsWorks Stacks is an application management service that provides an integrated experience for overseeing the complete application lifecycle. For information about this product, go to the AWS OpsWorks details page.

SDKs and CLI

The most common way to use the AWS OpsWorks Stacks API is by using the AWS Command Line Interface (CLI) or by using one of the AWS SDKs to implement applications in your preferred language. For more information, see:

- AWS CLI
- AWS SDK for Java
- AWS SDK for .NET
- AWS SDK for PHP 2
- AWS SDK for Ruby
- AWS SDK for Node.js
- AWS SDK for Python(Boto)

Endpoints

AWS OpsWorks Stacks supports the following endpoints, all HTTPS. You must connect to one of the following endpoints. Stacks can only be accessed or managed within the endpoint in which they are created.

- opsworks.us-east-1.amazonaws.com
- opsworks.us-east-2.amazonaws.com
- opsworks.us-west-1.amazonaws.com
- opsworks.us-west-2.amazonaws.com
- opsworks.ca-central-1.amazonaws.com (API only; not available in the AWS console)
- opsworks.eu-west-1.amazonaws.com
- opsworks.eu-west-2.amazonaws.com
- opsworks.eu-west-3.amazonaws.com
- opsworks.eu-central-1.amazonaws.com
- opsworks.ap-northeast-1.amazonaws.com
- opsworks.ap-northeast-2.amazonaws.com
- opsworks.ap-south-1.amazonaws.com
opsworks

- opsworks.ap-southeast-1.amazonaws.com
- opsworks.ap-southeast-2.amazonaws.com
- opsworks.sa-east-1.amazonaws.com

**Chef Versions**
When you call `create_stack`, `clone_stack`, or `update_stack` we recommend you use the `ConfigurationManager` parameter to specify the Chef version. The recommended and default value for Linux stacks is currently 12. Windows stacks use Chef 12.2. For more information, see **Chef Versions**.
You can specify Chef 12, 11.10, or 11.4 for your Linux stack. We recommend migrating your existing Linux stacks to Chef 12 as soon as possible.

**Usage**

```bash
opsworks(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

**Arguments**

- `config`: Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
    - `creds`:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
    - `anonymous`: Set anonymous credentials.
  - `endpoint`: The complete URL to use for the constructed client.
  - `region`: The AWS Region used in instantiating the client.
  - `close_connection`: Immediately close all HTTP connections.
  - `timeout`: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - `s3_force_path_style`: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
  - `sts_regional_endpoint`: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ent.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ent.html)

- `credentials`: Optional credentials shorthand for the `config` parameter
  - `creds`:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - `profile`: The name of a profile to use. If not given, then the default profile is used.
  - `anonymous`: Set anonymous credentials.

- `endpoint`: Optional shorthand for complete URL to use for the constructed client.
- `region`: Optional shorthand for AWS Region used in instantiating the client.
Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- opsworks(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `assign_instance`: Assign a registered instance to a layer
- `assign_volume`: Assigns one of the stack’s registered Amazon EBS volumes to a specified instance
- `associate_elastic_ip`: Associates one of the stack’s registered Elastic IP addresses with a specified instance
- `attach_elastic_load_balancer`: Attaches an Elastic Load Balancing load balancer to a specified layer
- `clone_stack`: Creates a clone of a specified stack
- `create_app`: Creates an app for a specified stack
- `create_deployment`: Runs deployment or stack commands
- `create_instance`: Creates an instance in a specified stack
<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>create_layer</td>
<td>Creates a layer</td>
</tr>
<tr>
<td>create_stack</td>
<td>Creates a new stack</td>
</tr>
<tr>
<td>create_user_profile</td>
<td>Creates a new user profile</td>
</tr>
<tr>
<td>delete_app</td>
<td>Deletes a specified app</td>
</tr>
<tr>
<td>delete_instance</td>
<td>Deletes a specified instance, which terminates the associated Amazon EC2 instance</td>
</tr>
<tr>
<td>delete_layer</td>
<td>Deletes a specified layer</td>
</tr>
<tr>
<td>delete_stack</td>
<td>Deletes a specified stack</td>
</tr>
<tr>
<td>delete_user_profile</td>
<td>Deletes a specified stack</td>
</tr>
<tr>
<td>deregister_ecs_cluster</td>
<td>Deregisters a specified Amazon ECS cluster from a stack</td>
</tr>
<tr>
<td>deregister_elastic_ip</td>
<td>Deregisters a specified Elastic IP address</td>
</tr>
<tr>
<td>deregister_instance</td>
<td>Deregister a registered Amazon EC2 or on-premises instance</td>
</tr>
<tr>
<td>deregister_rds_db_instance</td>
<td>Deregisters an Amazon RDS instance</td>
</tr>
<tr>
<td>deregister_volume</td>
<td>Deregisters an Amazon EBS volume</td>
</tr>
<tr>
<td>describe_agent_versions</td>
<td>Describes the available AWS OpsWorks Stacks agent versions</td>
</tr>
<tr>
<td>describe_apps</td>
<td>Requests a description of a specified set of apps</td>
</tr>
<tr>
<td>describe_commands</td>
<td>Describes the results of specified commands</td>
</tr>
<tr>
<td>describe_deployments</td>
<td>Requests a description of a specified set of deployments</td>
</tr>
<tr>
<td>describe_ecs_clusters</td>
<td>Describes Amazon ECS clusters that are registered with a stack</td>
</tr>
<tr>
<td>describe_elastic_ips</td>
<td>Describes Elastic IP addresses</td>
</tr>
<tr>
<td>describe_elastic_load_balancers</td>
<td>Describes a stack’s Elastic Load Balancing instances</td>
</tr>
<tr>
<td>describe_instances</td>
<td>Requests a description of a set of instances</td>
</tr>
<tr>
<td>describe_layers</td>
<td>Requests a description of one or more layers in a specified stack</td>
</tr>
<tr>
<td>describe_load_based_auto_scaling</td>
<td>Describes load-based auto scaling configurations for specified layers</td>
</tr>
<tr>
<td>describe_my_user_profile</td>
<td>Describes a user’s SSH information</td>
</tr>
<tr>
<td>describe_operating_systems</td>
<td>Describes the operating systems that are supported by AWS OpsWorks Stacks</td>
</tr>
<tr>
<td>describe_permissions</td>
<td>Describes the permissions for a specified stack</td>
</tr>
<tr>
<td>describe_raid_arrays</td>
<td>Describe an instance’s RAID arrays</td>
</tr>
<tr>
<td>describe_rds_db_instances</td>
<td>Describes Amazon RDS instances</td>
</tr>
<tr>
<td>describe_service_errors</td>
<td>Describes AWS OpsWorks Stacks service errors</td>
</tr>
<tr>
<td>describe_stack_provisioning_parameters</td>
<td>Requests a description of a stack’s provisioning parameters</td>
</tr>
<tr>
<td>describe_stacks</td>
<td>Requests a description of one or more stacks</td>
</tr>
<tr>
<td>describe_stack_summary</td>
<td>Describes the number of layers and apps in a specified stack, and the number of instances</td>
</tr>
<tr>
<td>describe_time_based_auto_scaling</td>
<td>Describes time-based auto scaling configurations for specified instances</td>
</tr>
<tr>
<td>describe_user_profiles</td>
<td>Describe specified users</td>
</tr>
<tr>
<td>describe_volumes</td>
<td>Describes an instance’s Amazon EBS volumes</td>
</tr>
<tr>
<td>detach_elastic_load_balancer</td>
<td>Detaches a specified Elastic Load Balancing instance from its layer</td>
</tr>
<tr>
<td>disassociate_elastic_ip</td>
<td>Disassociates an Elastic IP address from its instance</td>
</tr>
<tr>
<td>get_hostname_suggestion</td>
<td>Gets a generated host name for the specified layer, based on the current host name</td>
</tr>
<tr>
<td>grant_access</td>
<td>This action can be used only with Windows stacks</td>
</tr>
<tr>
<td>list_tags</td>
<td>Returns a list of tags that are applied to the specified stack or layer</td>
</tr>
<tr>
<td>reboot_instance</td>
<td>Reboots a specified instance</td>
</tr>
<tr>
<td>register_ecs_cluster</td>
<td>Registers a specified Amazon ECS cluster with a stack</td>
</tr>
<tr>
<td>register_elastic_ip</td>
<td>Registers an Elastic IP address with a specified stack</td>
</tr>
<tr>
<td>register_instance</td>
<td>Registers instances that were created outside of AWS OpsWorks Stacks with a specified stack</td>
</tr>
<tr>
<td>register_rds_db_instance</td>
<td>Registers an Amazon RDS instance with a stack</td>
</tr>
<tr>
<td>register_volume</td>
<td>Registers an Amazon EBS volume with a specified stack</td>
</tr>
<tr>
<td>set_load_based_auto_scaling</td>
<td>Specify the load-based auto scaling configuration for a specified layer</td>
</tr>
<tr>
<td>set_permission</td>
<td>Specifies a user’s permissions</td>
</tr>
</tbody>
</table>
set_time_based_auto_scaling
start_instance
start_stack
stop_instance
stop_stack
tag_resource
unassign_instance
unassign_volume
untag_resource
update_app
update_elastic_ip
update_instance
update_layer
update_my_user_profile
update_rds_db_instance
update_stack
update_user_profile
update_volume

Specify the time-based auto scaling configuration for a specified instance
Starts a specified instance
Starts a stack’s instances
Stops a specified instance
Stops a specified stack
Apply cost-allocation tags to a specified stack or layer in AWS OpsWorks Stacks
Unassigns a registered instance from all layers that are using the instance
Unassigns an assigned Amazon EBS volume
Removes tags from a specified stack or layer
Updates a specified app
Updates a registered Elastic IP address’s name
Updates a specified instance
Updates a specified layer
Updates a user’s SSH public key
Updates an Amazon RDS instance
Updates a specified stack
Updates a specified user profile
Updates an Amazon EBS volume’s name or mount point

Examples

```r
## Not run:
svc <- opsworks()
svc$assign_instance(
   Foo = 123
)

## End(Not run)
```

---

opsworkscm 

AWS OpsWorks CM

Description

AWS OpsWorks for configuration management (CM) is a service that runs and manages configuration management servers. You can use AWS OpsWorks CM to create and manage AWS OpsWorks for Chef Automate and AWS OpsWorks for Puppet Enterprise servers, and add or remove nodes for the servers to manage.

Glossary of terms

- **Server**: A configuration management server that can be highly-available. The configuration management server runs on an Amazon Elastic Compute Cloud (EC2) instance, and may use various other AWS services, such as Amazon Relational Database Service (RDS) and Elastic Load Balancing. A server is a generic abstraction over the configuration manager that you want to use, much like Amazon RDS. In AWS OpsWorks CM, you do not start or stop servers. After you create servers, they continue to run until they are deleted.
- **Engine**: The engine is the specific configuration manager that you want to use. Valid values in this release include ChefAutomate and Puppet.

- **Backup**: This is an application-level backup of the data that the configuration manager stores. AWS OpsWorks CM creates an S3 bucket for backups when you launch the first server. A backup maintains a snapshot of a server’s configuration-related attributes at the time the backup starts.

- **Events**: Events are always related to a server. Events are written during server creation, when health checks run, when backups are created, when system maintenance is performed, etc. When you delete a server, the server’s events are also deleted.

- **Account attributes**: Every account has attributes that are assigned in the AWS OpsWorks CM database. These attributes store information about configuration limits (servers, backups, etc.) and your customer account.

**Endpoints**

AWS OpsWorks CM supports the following endpoints, all HTTPS. You must connect to one of the following endpoints. Your servers can only be accessed or managed within the endpoint in which they are created.

- opsworks-cm.us-east-1.amazonaws.com
- opsworks-cm.us-east-2.amazonaws.com
- opsworks-cm.us-west-1.amazonaws.com
- opsworks-cm.us-west-2.amazonaws.com
- opsworks-cm.ap-northeast-1.amazonaws.com
- opsworks-cm.ap-southeast-1.amazonaws.com
- opsworks-cm.ap-southeast-2.amazonaws.com
- opsworks-cm.eu-central-1.amazonaws.com
- opsworks-cm.eu-west-1.amazonaws.com

For more information, see [AWS OpsWorks endpoints and quotas](https://aws.amazon.com/documentation/opsworks/#endpoints_and_quotas) in the AWS General Reference.

**Throttling limits**

All API operations allow for five requests per second with a burst of 10 requests per second.

**Usage**

```r
opsworkscm(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

**Arguments**

- `config` Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
- **creds:**
  - **access_key_id:** AWS access key ID
  - **secret_access_key:** AWS secret access key
  - **session_token:** AWS temporary session token
- **profile:** The name of a profile to use. If not given, then the default profile is used.
- **anonymous:** Set anonymous credentials.

  - **endpoint:** The complete URL to use for the constructed client.
  - **region:** The AWS Region used in instantiating the client.
  - **close_connection:** Immediately close all HTTP connections.
  - **timeout:** The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style:** Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  - **sts_regional_endpoint:** Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html

**credentials**  
Optional credentials shorthand for the config parameter

- **creds:**
  - **access_key_id:** AWS access key ID
  - **secret_access_key:** AWS secret access key
  - **session_token:** AWS temporary session token
- **profile:** The name of a profile to use. If not given, then the default profile is used.
- **anonymous:** Set anonymous credentials.

**endpoint**  
Optional shorthand for complete URL to use for the constructed client.

**region**  
Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- opsworkscm(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
    ),
  ),
```
anonymous = "logical"
),
endpoint = "string",
region = "string",
close_connection = "logical",
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

associate_node Associates a new node with the server
create_backup Creates an application-level backup of a server
create_server Creates and immediately starts a new server
delete_backup Deletes a backup
delete_server Deletes the server and the underlying AWS CloudFormation stacks (including the server's EC2 instance)
describe_account_attributes Describes your OpsWorks-CM account attributes
describe_backups Describes backups
describe_events Returns events for a specified server
describe_node_association_status Returns the current status of an existing association or disassociation request
describe_servers Lists all configuration management servers that are identified with your account
disable_node Disassociates a node from an AWS OpsWorks CM server, and removes the node from the server
export_server_engine_attribute Exports a specified server engine attribute as a base64-encoded string
list_tags_for_resource Returns a list of tags that are applied to the specified AWS OpsWorks for Chef Automate or AWS OpsWorks for Puppet Enterprise server
restore_server Restores a backup to a server that is in a CONNECTION_LOST, HEALTHY, RUNNING, UNHEALTHY, or TERMINATED state
start_maintenance Manually starts server maintenance
tag_resource Applies tags to an AWS OpsWorks for Chef Automate or AWS OpsWorks for Puppet Enterprise server or backup
untag_resource Removes specified tags from an AWS OpsWorks-CM server or backup
update_server Updates settings for a server
update_server_engine_attributes Updates engine-specific attributes on a specified server
Examples

```r
## Not run:
svc <- opsworkscm()
svc$associate_node(
    Foo = 123
)
## End(Not run)
```

---

**organizations**

**AWS Organizations**

---

**Description**

Organizations is a web service that enables you to consolidate your multiple Amazon Web Services accounts into an organization and centrally manage your accounts and their resources.

This guide provides descriptions of the Organizations operations. For more information about using this service, see the Organizations User Guide.

**Support and feedback for Organizations**

We welcome your feedback. Send your comments to feedback-awsorganizations@amazon.com or post your feedback and questions in the Organizations support forum. For more information about the Amazon Web Services support forums, see Forums Help.

**Endpoint to call When using the CLI or the Amazon Web Services SDK**

For the current release of Organizations, specify the us-east-1 region for all Amazon Web Services API and CLI calls made from the commercial Amazon Web Services Regions outside of China. If calling from one of the Amazon Web Services Regions in China, then specify cn-northwest-1. You can do this in the CLI by using these parameters and commands:

- Use the following parameter with each command to specify both the endpoint and its region:
  ```
  --endpoint-url https://organizations.us-east-1.amazonaws.com
  ```
  (from commercial Amazon Web Services Regions outside of China)
  or
  ```
  --endpoint-url https://organizations.cn-northwest-1.amazonaws.com.cn
  ```
  (from Amazon Web Services Regions in China)
- Use the default endpoint, but configure your default region with this command:
  ```
  aws configure set default.region us-east-1
  ```
  (from commercial Amazon Web Services Regions outside of China)
  or
  ```
  aws configure set default.region cn-northwest-1
  ```
  (from Amazon Web Services Regions in China)
- Use the following parameter with each command to specify the endpoint:
  ```
  --region us-east-1
  ```
  (from commercial Amazon Web Services Regions outside of China)
  or
  ```
  --region cn-northwest-1
  ```
  (from Amazon Web Services Regions in China)
Recording API Requests

Organizations supports CloudTrail, a service that records Amazon Web Services API calls for your Amazon Web Services account and delivers log files to an Amazon S3 bucket. By using information collected by CloudTrail, you can determine which requests the Organizations service received, who made the request and when, and so on. For more about Organizations and its support for CloudTrail, see Logging Organizations API calls with CloudTrail in the Organizations User Guide. To learn more about CloudTrail, including how to turn it on and find your log files, see the CloudTrail User Guide.

Usage

```r
organizations(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>config</td>
<td>Optional configuration of credentials, endpoint, and/or region.</td>
</tr>
<tr>
<td>credentials</td>
<td>Optional credentials shorthand for the config parameter</td>
</tr>
<tr>
<td>endpoint</td>
<td>The complete URL to use for the constructed client.</td>
</tr>
<tr>
<td>region</td>
<td>The AWS Region used in instantiating the client.</td>
</tr>
<tr>
<td>close_connection</td>
<td>Immediately close all HTTP connections.</td>
</tr>
<tr>
<td>timeout</td>
<td>The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.</td>
</tr>
<tr>
<td>s3_force_path_style</td>
<td>Set this to true to force the request to use path-style addressing, i.e. <a href="http://s3.amazonaws.com/BUCKET/KEY">http://s3.amazonaws.com/BUCKET/KEY</a>.</td>
</tr>
<tr>
<td>sts_regional_endpoint</td>
<td>Set sts regional endpoint resolver to regional or legacy <a href="https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-en.html">https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-en.html</a></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>credentials</td>
<td>Optional credentials shorthand for the config parameter</td>
</tr>
<tr>
<td></td>
<td>• credss:</td>
</tr>
<tr>
<td></td>
<td>• access_key_id: AWS access key ID</td>
</tr>
<tr>
<td></td>
<td>• secret_access_key: AWS secret access key</td>
</tr>
<tr>
<td></td>
<td>• session_token: AWS temporary session token</td>
</tr>
<tr>
<td></td>
<td>• profile: The name of a profile to use. If not given, then the default profile is used.</td>
</tr>
<tr>
<td></td>
<td>• anonymous: Set anonymous credentials.</td>
</tr>
<tr>
<td></td>
<td>• endpoint: The complete URL to use for the constructed client.</td>
</tr>
<tr>
<td></td>
<td>• region: The AWS Region used in instantiating the client.</td>
</tr>
<tr>
<td></td>
<td>• close_connection: Immediately close all HTTP connections.</td>
</tr>
<tr>
<td></td>
<td>• timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.</td>
</tr>
<tr>
<td></td>
<td>• s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. <a href="http://s3.amazonaws.com/BUCKET/KEY">http://s3.amazonaws.com/BUCKET/KEY</a>.</td>
</tr>
<tr>
<td></td>
<td>• sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy <a href="https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-en.html">https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-en.html</a></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• creds:</td>
</tr>
<tr>
<td></td>
<td>• access_key_id: AWS access key ID</td>
</tr>
<tr>
<td></td>
<td>• secret_access_key: AWS secret access key</td>
</tr>
<tr>
<td></td>
<td>• session_token: AWS temporary session token</td>
</tr>
<tr>
<td></td>
<td>• profile: The name of a profile to use. If not given, then the default profile is used.</td>
</tr>
</tbody>
</table>
organizations

- **anonymous**: Set anonymous credentials.

  endpoint       Optional shorthand for complete URL to use for the constructed client.

  region        Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...),` where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- organizations(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

**Operations**

- `accept_handshake` Sends a response to the originator of a handshake agreeing to the action proposed
- `attach_policy` Attaches a policy to a root, an organizational unit (OU), or an individual account
- `cancel_handshake` Cancels a handshake
organizations

- `close_account`: Closes an Amazon Web Services member account within an organization.
- `create_account`: Creates an Amazon Web Services account that is automatically a member of the organization whose credentials made the request.
- `create_gov_cloud_account`: This action is available if all of the following are true:
- `create_organization`: Creates an Amazon Web Services organization.
- `create_organizational_unit`: Creates an organizational unit (OU) within a root or parent OU.
- `create_policy`: Creates a policy of a specified type that you can attach to a root, an organizational unit (OU), or an individual Amazon Web Services account.
- `decline_handshake`: Declines a handshake request.
- `delete_account`: Deletes the organization.
- `delete_organizational_unit`: Deletes an organizational unit (OU) from a root or another OU.
- `delete_policy`: Deletes the specified policy from your organization.
- `delete_resource_policy`: Deletes the resource policy from your organization.
- `deregister_delegated_administrator`: Removes the specified member Amazon Web Services account as a delegated administrator for the specified Amazon Web Services service.
- `describe_account`: Retrieves Organizations-related information about the specified account.
- `describe_create_account_status`: Retrieves the current status of an asynchronous request to create an account.
- `describe_effective_policy`: Returns the contents of the effective policy for specified policy type and account.
- `describe_handshake`: Retrieves information about a previously requested handshake.
- `describe_organization`: Retrieves information about the organization that the user's account belongs to.
- `describe_organizational_unit`: Retrieves information about an organizational unit (OU).
- `describe_policy`: Retrieves information about a policy.
- `describe_resource_policy`: Retrieves information about a resource policy.
- `detach_policy`: Detaches a policy from a target root, organizational unit (OU), or account.
- `disable_aws_service_access`: Disables the integration of an Amazon Web Services service (the service that is specified by ServicePrincipal) with Organizations.
- `disable_policy_type`: Disables an organizational policy type in a root.
- `enable_all_features`: Enables all features in an organization.
- `enable_aws_service_access`: Enables the integration of an Amazon Web Services service (the service that is specified by ServicePrincipal) with Organizations.
- `enable_policy_type`: Enables a policy type in a root.
- `invite_account_to_organization`: Sends an invitation to another account to join your organization as a member account.
- `leave_organization`: Removes a member account from its parent organization.
- `list_accounts`: Lists all the accounts in the organization.
- `list_accounts_for_parent`: Lists the accounts in an organization that are contained by the specified target root or organizational unit (OU).
- `list_aws_service_access_for_organization`: Lists the Amazon Web Services services that you enabled to integrate with Organizations.
- `list_children`: Lists all of the organizational units (OUs) or accounts that are contained in the specified organization.
- `list_create_account_status`: Lists the account creation requests that match the specified status that is currently being tracked for the organization.
- `list_delegated_administrators`: Lists the Amazon Web Services accounts that are designated as delegated administrators in an organization.
- `list_delegated_services_for_account`: Lists the Amazon Web Services services for which the specified account is a delegated administrator.
- `list_effective_policies`: Lists the current handshakes that are associated with the account of the requesting user.
- `list_effective_policies_for_organizational_unit`: Lists the policies that are associated with the organization that the requesting user is a member of.
- `list_organizational_units_for_parent`: Lists the organizational units (OUs) in a parent organizational unit or root.
- `list_organizational_units_for_parent`: Lists the root or organizational units (OUs) that serve as the immediate parent of the specified child OU or account.
- `list_parents`: Lists the roots that are defined in the current organization.
- `list_tags_for_resource`: Lists tags that are attached to the specified resource.
- `list_aws_service_access_for_organizational_unit`: Lists all the roots, organizational units (OUs), and accounts that the specified policy type is attached to.
- `move_account`: Moves an account from its current source parent root or organizational unit (OU).
- `put_resource_policy`: Creates or updates a resource policy.
- `register_delegated_administrator`: Enables the specified member account to administer the Organizations features of the specified Amazon Web Services service.
- `remove_account_from_organization`: Removes the specified account from the organization.
**Examples**

```r
# Not run:
svc <- organizations()
# Bill is the owner of an organization, and he invites Juan's account
# (222222222222) to join his organization. The following example shows
# Juan's account accepting the handshake and thus agreeing to the
# invitation.
svc$accept_handshake(
  HandshakeId = "h-examplehandshakeid111"
)
```

---

**Description**

**Overview**

This is the **AWS Panorama API Reference**. For an introduction to the service, see *What is AWS Panorama?* in the [AWS Panorama Developer Guide](https://aws.amazon.com/panorama/).

**Usage**

```r
panorama(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

**Arguments**

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
• **endpoint**: The complete URL to use for the constructed client.
• **region**: The AWS Region used in instantiating the client.
• **close_connection**: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-...html

**credentials**

Optional credentials shorthand for the config parameter

• **creds**:
  – **access_key_id**: AWS access key ID
  – **secret_access_key**: AWS secret access key
  – **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.
• **anonymous**: Set anonymous credentials.

**endpoint**

Optional shorthand for complete URL to use for the constructed client.

**region**

Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- panorama(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  )
)
```
),
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

create_application_instance Creates an application instance and deploys it to a device
create_job_for_devices Creates a job to run on a device
create_node_from_template_job Creates a camera stream node
create_package Creates a package and storage location in an Amazon S3 access point
create_package_import_job Imports a node package
delete_device Deletes a device
delete_package Deletes a package
deregister_package_version Deregisters a package version
describe_application_instance Returns information about an application instance on a device
describe_application_instance_details Returns information about an application instance’s configuration manifest
describe_device Returns information about a device
describe_device_job Returns information about a device job
describe_node Returns information about a node
describe_node_from_template_job Returns information about a job to create a camera stream node
describe_package Returns information about a package
describe_package_import_job Returns information about a package import job
describe_package_version Returns information about a package version
list_application_instance_dependencies Returns a list of application instance dependencies
list_application_instance_node_instances Returns a list of application node instances
list_application_instances Returns a list of application instances
list Devices
list_devices_jobs
list_node_from_template_jobs
list_nodes
list_package_import_jobs
list_packages
list_tags_for_resource
provision_device
register_package_version
remove_application_instance
signal_application_instance_node_instances
tag_resource
Tags a resource
untag_resource
update_device_metadata

Examples

```r
## Not run:
svc <- panorama()
svc$create_application_instance(
  Foo = 123
)

## End(Not run)
```

---

### Description

Amazon Web Services Payment Cryptography Control Plane APIs manage encryption keys for use during payment-related cryptographic operations. You can create, import, export, share, manage, and delete keys. You can also manage Identity and Access Management (IAM) policies for keys. For more information, see Identity and access management in the Amazon Web Services Payment Cryptography User Guide.

To use encryption keys for payment-related transaction processing and associated cryptographic operations, you use the Amazon Web Services Payment Cryptography Data Plane. You can perform actions like encrypt, decrypt, generate, and verify payment-related data.

All Amazon Web Services Payment Cryptography API calls must be signed and transmitted using Transport Layer Security (TLS). We recommend you always use the latest supported TLS version for logging API requests.

Amazon Web Services Payment Cryptography supports CloudTrail for control plane operations, a service that logs Amazon Web Services API calls and related events for your Amazon Web Services account and delivers them to an Amazon S3 bucket you specify. By using the information collected by CloudTrail, you can determine what requests were made to Amazon Web Services Payment Cryptography, who made the request, when it was made, and so on. If you don’t configure a trail, you can still view the most recent events in the CloudTrail console. For more information, see the CloudTrail User Guide.

### Usage

```r
cryptographycontrolplane(
  config = list(),
  credentials = list(),
)```
Arguments

Optional configuration of credentials, endpoint, and/or region.

- **credentials**:
  - **creds**:
    * **access_key_id**: AWS access key ID
    * **secret_access_key**: AWS secret access key
    * **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

- **endpoint**: The complete URL to use for the constructed client.

- **region**: The AWS Region used in instantiating the client.

- **close_connection**: Immediately close all HTTP connections.

- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

- **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```r
svc <- paymentcryptographycontrolplane(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- **create_alias**: Creates an alias, or a friendly name, for an Amazon Web Services Payment Cryptography key
- **create_key**: Creates an Amazon Web Services Payment Cryptography key, a logical representation of a cryptographic key
- **delete_alias**: Deletes the alias, but doesn’t affect the underlying key
- **delete_key**: Deletes the key material and metadata associated with Amazon Web Services Payment Cryptography
- **export_key**: Exports a key from Amazon Web Services Payment Cryptography
- **get_alias**: Gets the Amazon Web Services Payment Cryptography key associated with the alias
- **get_key**: Gets the key material for an Amazon Web Services Payment Cryptography key, including the import token and the wrapping key certificate in PEM format (base64 encoded) to initiate a TR-34 WrappedKeyBlock or a RSA WrappedKeyCryptogram import into Amazon Web Services Payment Cryptography
- **get_parameters_for_export**: Gets the export token and the signing key certificate to initiate a TR-34 key export from Amazon Web Services Payment Cryptography
- **get_parameters_for_import**: Gets the import token and the wrapping key certificate in PEM format (base64 encoded) to initiate a TR-34 WrappedKeyBlock or a RSA WrappedKeyCryptogram import into Amazon Web Services Payment Cryptography
- **get_public_key_certificate**: Gets the public key certificate of the asymmetric key pair that exists within Amazon Web Services Payment Cryptography
- **import_key**: Imports symmetric keys and public key certificates in PEM format (base64 encoded) into Amazon Web Services Payment Cryptography
- **list_aliases**: Lists the aliases for all keys in the caller’s Amazon Web Services account and Amazon Web Services Region
- **list_keys**: Lists the keys in the caller’s Amazon Web Services account and Amazon Web Services Region
- **list_tags_for_resource**: Lists the tags for an Amazon Web Services resource
Description

You use the Amazon Web Services Payment Cryptography Data Plane to manage how encryption keys are used for payment-related transaction processing and associated cryptographic operations. You can encrypt, decrypt, generate, verify, and translate payment-related cryptographic operations in Amazon Web Services Payment Cryptography. For more information, see Data operations in the Amazon Web Services Payment Cryptography User Guide.

To manage your encryption keys, you use the Amazon Web Services Payment Cryptography Control Plane. You can create, import, export, share, manage, and delete keys. You can also manage Identity and Access Management (IAM) policies for keys.

Usage

```r
paymentcryptographydataplane(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```
Arguments

config Optional configuration of credentials, endpoint, and/or region.

- credentials:
  - creds:
    - access_key_id: AWS access key ID
    - secret_access_key: AWS secret access key
    - session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.

- endpoint: The complete URL to use for the constructed client.

- region: The AWS Region used in instantiating the client.

- close_connection: Immediately close all HTTP connections.

- timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

- s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

- sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter

- creds:
  - access_key_id: AWS access key ID
  - secret_access_key: AWS secret access key
  - session_token: AWS temporary session token

- profile: The name of a profile to use. If not given, then the default profile is used.

- anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- paymentcryptographydataplane(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ))
  ))
```
secret_access_key = "string",
session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string",
close_connection = "logical",
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
creds = list(  
access_key_id = "string",
secret_access_key = "string",
session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

decrypt_data
encrypt_data
generate_card_validation_data
generate_mac
generate_pin_data
re_encrypt_data
translate_pin_data
verify_auth_request_cryptogram
verify_card_validation_data
verify_mac
verify_pin_data

Decrypts ciphertext data to plaintext using a symmetric (TDES, AES), asymmetric (RSA), or derived (DUKPT or EMV) encryption key scheme
Encrypts plaintext data to ciphertext using a symmetric (TDES, AES), asymmetric (RSA), or derived (DUKPT or EMV) encryption key scheme
Generates card-related validation data using algorithms such as Card Verification Values (CVV/CVV2), Dynamic Card Verification Values (dCVV/dCVV2), or Card Security Codes (CSC)
Generates a Message Authentication Code (MAC) cryptogram within Amazon Web Services
Generates pin-related data such as PIN, PIN Verification Value (PVV), PIN Block, and PIN Offset during new card issuance or reissuance
Re-encrypt ciphertext using DUKPT, Symmetric and Asymmetric Data Encryption Keys
Translates encrypted PIN block from and to ISO 9564 formats 0,1,3,4
Verifies Authorization Request Cryptogram (ARQC) for an EMV chip payment card authorization
Verifies card-related validation data using algorithms such as Card Verification Values (CVV/CVV2), Dynamic Card Verification Values (dCVV/dCVV2) and Card Security Codes (CSC)
Verifies a Message Authentication Code (MAC)
Verifies pin-related data such as PIN and PIN Offset using algorithms including VISA PVV

Examples

## Not run:
svc <- paymentcryptographydataplane()
svc$decrypt_data(
  Foo = 123
)
## End(Not run)

### pcaconnectorad

**PcaConnectorAd**

## Description

Amazon Web Services Private CA Connector for Active Directory creates a connector between Amazon Web Services Private CA and Active Directory (AD) that enables you to provision security certificates for AD signed by a private CA that you own. For more information, see Amazon Web Services Private CA Connector for Active Directory.

## Usage

```r
pcaconnectorad(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

## Arguments

**config** Optional configuration of credentials, endpoint, and/or region.

- **credentials**:
  - **creds**:
    * access_key_id: AWS access key ID
    * secret_access_key: AWS secret access key
    * session_token: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.
- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html
credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value
A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax
```
svc <- pcaconnectorad(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
```
Operations

create_connector
create_directory_registration
create_service_principal_name
create_template
create_template_group_access_control_entry
delete_connector
delete_directory_registration
delete_service_principal_name
delete_template
delete_template_group_access_control_entry
get_connector
get_directory_registration
get_service_principal_name
get_template
get_template_group_access_control_entry
list_connectors
list_directory_registrations
list_service_principal_names
list_tags_for_resource
list_template_group_access_control_entries
list_templates
tag_resource
untag_resource
update_template
update_template_group_access_control_entry

Create a connector between Amazon Web Services Private CA and an Active Directory.
Creates a directory registration that authorizes communication between Amazon Web Services Private CA and an Active Directory.
Creates a service principal name (SPN) for the service account in Active Directory.
Creates an Active Directory compatible certificate template.
Create a group access control entry.
Deletes a connector for Active Directory.
Deletes a directory registration.
Deletes the service principal name (SPN) used by a connector to authenticate with Active Directory.
Deletes a template.
Deletes a group access control entry.
Lists information about your connector.
A structure that contains information about your directory registration.
Lists the service principal name that the connector uses to authenticate with Active Directory.
Retrieves a certificate template that the connector uses to issue certificates from.
Retrieves the group access control entries for a template.
Lists the connectors that you created by using the https://docs
Lists the directory registrations that you created by using the https://docs
Lists the service principal names that the connector uses to authenticate with Active Directory.
Lists the tags, if any, that are associated with your resource.
Lists group access control entries you created.
Lists the templates, if any, that are associated with a connector.
Adds one or more tags to your resource.
Removes one or more tags from your resource.
Update template configuration to define the information included in certificates.
Update a group access control entry you created using CreateTemplateGroupAccessControlEntry.

Examples

```r
## Not run:
svc <- pcaconnectorad()
svc$create_connector(
   Foo = 123
)

## End(Not run)
```

personalize  Amazon Personalize
Description

Amazon Personalize is a machine learning service that makes it easy to add individualized recommendations to customers.

Usage

personalize(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

config Optional configuration of credentials, endpoint, and/or region.

  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.

  • endpoint: The complete URL to use for the constructed client.

  • region: The AWS Region used in instantiating the client.

  • close_connection: Immediately close all HTTP connections.

  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy http://s3.amazonaws.com/BUCKET/KEY.

credentials Optional credentials shorthand for the config parameter

  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token

  • profile: The name of a profile to use. If not given, then the default profile is used.

  • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.
Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- personalize(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"），
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `create_batch_inference_job`: Generates batch recommendations based on a list of items or users stored in Amazon S3 and exports them to an Amazon S3 bucket.
- `create_batch_segment_job`: Creates a batch segment job.
- `create_campaign`: You incur campaign costs while it is active.
- `create_dataset`: Creates an empty dataset and adds it to the specified dataset group.
- `create_dataset_export_job`: Creates a job that exports data from your dataset to an Amazon S3 bucket.
- `create_dataset_group`: Creates an empty dataset group.
- `create_dataset_import_job`: Creates a job that imports training data from your data source (an Amazon S3 bucket) to an Amazon Personalize dataset.
- `create_event_tracker`: Creates an event tracker that you use when adding event data to a specified dataset group using the PutEvents API.
list_solution_versions  Returns a list of solution versions for the given solution
list_tags_for_resource  Get a list of tags attached to a resource
start_recommender  Starts a recommender that is INACTIVE
stop_recommender  Stops a recommender that is ACTIVE
stop_solution_version_creation  Stops creating a solution version that is in a state of CREATE_PENDING or CREATE_IN_PROGRESS
tag_resource  Add a list of tags to a resource
untag_resource  Removes the specified tags that are attached to a resource
update_campaign  Updates a campaign to deploy a retrained solution version with an existing campaign, change your campaign’s minProvisionedTPS, or modify your campaign’s configuration
update_dataset  Update a dataset to replace its schema with a new or existing one
update_meter_attribution  Updates a metric attribution
update_recommender  Updates the recommender to modify the recommender configuration

Examples

```r
## Not run:
svc <- personalize()
svc$select_batch_inference_job(
  Foo = 123
)
## End(Not run)
```

---

**personalizeevents**  *Amazon Personalize Events*

**Description**

Amazon Personalize can consume real-time user event data, such as stream or click data, and use it for model training either alone or combined with historical data. For more information see **Recording item interaction events**.

**Usage**

```r
personalizeevents(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

**Arguments**

- `config`  Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
- **creds:**
  - **access_key_id:** AWS access key ID
  - **secret_access_key:** AWS secret access key
  - **session_token:** AWS temporary session token

- **profile:** The name of a profile to use. If not given, then the default profile is used.

- **anonymous:** Set anonymous credentials.

**endpoint**: The complete URL to use for the constructed client.

**region**: The AWS Region used in instantiating the client.

**close_connection**: Immediately close all HTTP connections.

**timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

**s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.

**sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy `https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-<html>`

---

**credentials** Optional credentials shorthand for the config parameter

- **creds:**
  - **access_key_id:** AWS access key ID
  - **secret_access_key:** AWS secret access key
  - **session_token:** AWS temporary session token

- **profile:** The name of a profile to use. If not given, then the default profile is used.

- **anonymous:** Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

---

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- personalizeevents(
  config = list(  
    credentials = list(  
      creds = list(  
        access_key_id = "string",  
        secret_access_key = "string",  
        session_token = "string"  
      ),  
      profile = "string",  
    ),  
  ),
)```
 anonymous = "logical"
),
 endpoint = "string",
 region = "string",
 close_connection = "logical",
 timeout = "numeric",
 s3_force_path_style = "logical",
 sts_regional_endpoint = "string"
),
 credentials = list(
 creds = list(
   access_key_id = "string",
   secret_access_key = "string",
   session_token = "string"
 ),
   profile = "string",
   anonymous = "logical"
 ),
 endpoint = "string",
 region = "string"
 )

Operations

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>put_action_interactions</td>
<td>Records action interaction event data</td>
</tr>
<tr>
<td>put_actions</td>
<td>Adds one or more actions to an Actions dataset</td>
</tr>
<tr>
<td>put_events</td>
<td>Records item interaction event data</td>
</tr>
<tr>
<td>put_items</td>
<td>Adds one or more items to an Items dataset</td>
</tr>
<tr>
<td>put_users</td>
<td>Adds one or more users to a Users dataset</td>
</tr>
</tbody>
</table>

Examples

```r
## Not run:
svc <- personalizeevents()
svc$put_action_interactions(
  Foo = 123
)
## End(Not run)
```

personalizeruntime Amazon Personalize Runtime
Description

Amazon Personalize Runtime

Usage

personalizeruntime(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

config Optional configuration of credentials, endpoint, and/or region.

• credentials:
  – creds:
    * access_key_id: AWS access key ID
    * secret_access_key: AWS secret access key
    * session_token: AWS temporary session token
  – profile: The name of a profile to use. If not given, then the default
    profile is used.
  – anonymous: Set anonymous credentials.
• endpoint: The complete URL to use for the constructed client.
• region: The AWS Region used in instantiating the client.
• close_connection: Immediately close all HTTP connections.
• timeout: The time in seconds till a timeout exception is thrown when at-
  tempting to make a connection. The default is 60 seconds.
• s3_force_path_style: Set this to true to force the request to use path-style
  addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• sts_regional_endpoint: Set sts regional endpoint resolver to regional or

credentials Optional credentials shorthand for the config parameter

• creds:
  – access_key_id: AWS access key ID
  – secret_access_key: AWS secret access key
  – session_token: AWS temporary session token
• profile: The name of a profile to use. If not given, then the default profile
  is used.
• anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.
Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- personalizeruntime(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `get_action_recommendations`: Returns a list of recommended actions in sorted in descending order by prediction score
- `get_personalized_ranking`: Re-ranks a list of recommended items for the given user
- `get_recommendations`: Returns a list of recommended items
Examples

```r
## Not run:
svc <- personalizeruntime()
svc$get_action_recommendations(
  Foo = 123
)
## End(Not run)
```

Description

Amazon RDS Performance Insights

Amazon RDS Performance Insights enables you to monitor and explore different dimensions of database load based on data captured from a running DB instance. The guide provides detailed information about Performance Insights data types, parameters and errors.

When Performance Insights is enabled, the Amazon RDS Performance Insights API provides visibility into the performance of your DB instance. Amazon CloudWatch provides the authoritative source for Amazon Web Services service-vended monitoring metrics. Performance Insights offers a domain-specific view of DB load.

DB load is measured as average active sessions. Performance Insights provides the data to API consumers as a two-dimensional time-series dataset. The time dimension provides DB load data for each time point in the queried time range. Each time point decomposes overall load in relation to the requested dimensions, measured at that time point. Examples include SQL, Wait event, User, and Host.

- To learn more about Performance Insights and Amazon Aurora DB instances, go to the Amazon Aurora User Guide.
- To learn more about Performance Insights and Amazon RDS DB instances, go to the Amazon RDS User Guide.
- To learn more about Performance Insights and Amazon DocumentDB clusters, go to the Amazon DocumentDB Developer Guide.

Usage

```r
pi(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

`config`  
Optional configuration of credentials, endpoint, and/or region.

- `credentials`:
  - `creds`:
* access_key_id: AWS access key ID
* secret_access_key: AWS secret access key
* session_token: AWS temporary session token

- profile: The name of a profile to use. If not given, then the default profile is used.

- anonymous: Set anonymous credentials.

- endpoint: The complete URL to use for the constructed client.
- region: The AWS Region used in instantiating the client.
- close_connection: Immediately close all HTTP connections.
- timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
- sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoint.html

credentials Optional credentials shorthand for the config parameter

- creds:
  - access_key_id: AWS access key ID
  - secret_access_key: AWS secret access key
  - session_token: AWS temporary session token

- profile: The name of a profile to use. If not given, then the default profile is used.

- anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- pi(
config = list(
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
```
),
endpoint = "string",
region = "string",
close_connection = "logical",
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

create_performance_analysis_report: creates a new performance analysis report for a specific time period for the DB instance.
delete_performance_analysis_report: deletes a performance analysis report.
describe_dimension_keys: for a specific time period, retrieve the top N dimension keys for a metric.
get_dimension_key_details: get the attributes of the specified dimension group for a DB instance or data source.
get_performance_analysis_report: retrieves the report including the report ID, status, time details, and the insights with recommendations.
get_resource_metadata: retrieve the metadata for different features.
get_resource_metrics: retrieve Performance Insights metrics for a set of data sources over a time period.
list_available_resource_dimensions: retrieve the dimensions that can be queried for each specified metric type on a specified DB instance.
list_available_resource_metrics: retrieve metrics of the specified types that can be queried for a specified DB instance.
list_performance_analysis_reports: lists all the analysis reports created for the DB instance.
list_tags_for_resource: retrieves all the metadata tags associated with Amazon RDS Performance Insights resource.
tag_resource: adds metadata tags to the Amazon RDS Performance Insights resource.
tag_resource: deletes the metadata tags from the Amazon RDS Performance Insights resource.

Examples

```r
## Not run:
svc <- pi()
svc$create_performance_analysis_report(
    Foo = 123
)
## End(Not run)
```
**Description**

Doc Engage API - Amazon Pinpoint API

**Usage**

```
pinpoint(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>config</code></td>
<td>Optional configuration of credentials, endpoint, and/or region.</td>
</tr>
<tr>
<td></td>
<td>• <code>credentials</code>:&lt;br&gt; - <code>creds</code>:&lt;br&gt; * <code>access_key_id</code>: AWS access key ID&lt;br&gt; * <code>secret_access_key</code>: AWS secret access key&lt;br&gt; * <code>session_token</code>: AWS temporary session token&lt;br&gt; - <code>profile</code>: The name of a profile to use. If not given, then the default profile is used.&lt;br&gt; - <code>anonymous</code>: Set anonymous credentials.&lt;br&gt; • <code>endpoint</code>: The complete URL to use for the constructed client.&lt;br&gt; • <code>region</code>: The AWS Region used in instantiating the client.&lt;br&gt; • <code>close_connection</code>: Immediately close all HTTP connections.&lt;br&gt; • <code>timeout</code>: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.&lt;br&gt; • <code>s3_force_path_style</code>: Set this to <code>true</code> to force the request to use path-style addressing, i.e. <code>http://s3.amazonaws.com/BUCKET/KEY</code>.&lt;br&gt; • <code>sts_regional_endpoint</code>: Set sts regional endpoint resolver to regional or legacy <a href="https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html">https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html</a></td>
</tr>
<tr>
<td><code>credentials</code></td>
<td>Optional credentials shorthand for the config parameter</td>
</tr>
<tr>
<td><code>endpoint</code></td>
<td>Optional shorthand for complete URL to use for the constructed client.</td>
</tr>
<tr>
<td><code>region</code></td>
<td>Optional shorthand for AWS Region used in instantiating the client.</td>
</tr>
</tbody>
</table>
Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- pinpoint(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `create_app`: Creates an application
- `create_campaign`: Creates a new campaign for an application or updates the settings of an existing campaign for an application
- `create_email_template`: Creates a message template for messages that are sent through the email channel
- `create_export_job`: Creates an export job for an application
- `create_import_job`: Creates an import job for an application
- `create_in_app_template`: Creates a new message template for messages using the in-app message channel
- `create_journey`: Creates a journey for an application
- `create_push_template`: Creates a message template for messages that are sent through a push notification channel
create_recommender_configuration
create_segment
create_sms_template
create_voice_template
delete_adm_channel
delete_apns_channel
delete_apns_sandbox_channel
delete_apns_voip_channel
delete_apns_voip_sandbox_channel
delete_app
delete_baidu_channel
delete_campaign
delete_email_channel
delete_email_template
delete_endpoint
delete_event_stream
delete_gcm_channel
delete_in_app_template
delete_journey
delete_push_template
delete_recommender_configuration
delete_segment
delete_sms_channel
delete_sms_template
delete_user_endpoints
delete_voice_channel
delete_voice_template
get_adm_channel
get_apns_channel
get_apns_sandbox_channel
get_apns_voip_channel
get_apns_voip_sandbox_channel
get_app
get_application_date_range_kpi
get_application_settings
get_apps
get_baidu_channel
get_campaign
get_campaign_activities
get_campaign_date_range_kpi
get_campaigns
get_campaign_version
get_campaign_versions
get_channels
get_email_channel
get_email_template
get_endpoint
get_event_stream

create_recommender_configuration
create_segment
create_sms_template
create_voice_template
delete_adm_channel
delete_apns_channel
delete_apns_sandbox_channel
delete_apns_voip_channel
delete_apns_voip_sandbox_channel
delete_app
delete_baidu_channel
delete_campaign
delete_email_channel
delete_email_template
delete_endpoint
delete_event_stream
delete_gcm_channel
delete_in_app_template
delete_journey
delete_push_template
delete_recommender_configuration
delete_segment
delete_sms_channel
delete_sms_template
delete_user_endpoints
delete_voice_channel
delete_voice_template
get_adm_channel
get_apns_channel
get_apns_sandbox_channel
get_apns_voip_channel
get_apns_voip_sandbox_channel
get_app
get_application_date_range_kpi
get_application_settings
get_apps
get_baidu_channel
get_campaign
get_campaign_activities
get_campaign_date_range_kpi
get_campaigns
get_campaign_version
get_campaign_versions
get_channels
get_email_channel
get_email_template
get_endpoint
get_event_stream

Create an Amazon Pinpoint configuration for a recommender model
Creates a new segment for an application or updates the configuration, dimensions, and segments
Creates a message template for messages that are sent through the SMS channel
Creates a message template for messages that are sent through the voice channel
Disables the ADM channel for an application and deletes any existing settings
Disables the APNs channel for an application and deletes any existing settings
Disables the APNs sandbox channel for an application and deletes any existing settings
Disables the APNs VoIP channel for an application and deletes any existing settings
Disables the APNs VoIP sandbox channel for an application and deletes any existing settings
Deletes an application
Disables the Baidu channel for an application and deletes any existing settings
Deletes a campaign from an application
Disables the email channel for an application and deletes any existing settings
Deletes a message template for messages that were sent through the email channel
Deletes an endpoint from an application
Deletes the event stream for an application
Disables the GCM channel for an application and deletes any existing settings
Deletes a message template for messages sent using the in-app message channel
Deletes a journey from an application
Deletes a message template for messages that were sent through a push notification channel
Deletes an Amazon Pinpoint configuration for a recommender model
Deletes a segment from an application
Disables the SMS channel for an application and deletes any existing settings
Deletes a message template for messages that were sent through the SMS channel
Deletes all the endpoints that are associated with a specific user ID
Disables the voice channel for an application and deletes any existing settings
Deletes a message template for messages that were sent through the voice channel
Retrieves information about the status and settings of the ADM channel for an application
Retrieves information about the status and settings of the APNs channel for an application
Retrieves information about the status and settings of the APNs sandbox channel for an application
Retrieves information about the status and settings of the APNs VoIP channel for an application
Retrieves information about the status and settings of the APNs VoIP sandbox channel for an application
Retrieves information about an application
Retrieves (queries) pre-aggregated data for a standard metric that applies to an application
Retrieves information about the settings for an application
Retrieves information about all the applications that are associated with your Amazon Pinpoint account
Retrieves information about the status and settings of the Baidu channel for an application
Retrieves information about the status, configuration, and other settings for a campaign
Retrieves information about all the activities for a campaign
Retrieves (queries) pre-aggregated data for a standard metric that applies to a campaign
Retrieves information about the status, configuration, and other settings for all campaigns
Retrieves information about the status, configuration, and other settings for a specific version of a campaign
Retrieves information about the status, configuration, and other settings for all versions of a campaign
Retrieves information about the history and status of each channel for an application
Retrieves information about the status and settings of the email channel for an application
Retrieves the content and settings of a message template for messages that are sent through the email channel
Retrieves information about the settings and attributes of a specific endpoint for an application
Retrieves information about the event stream settings for an application
get_export_job
get_export_jobs
get_gcm_channel
get_import_job
get_import_jobs
get_in_app_messages
get_in_app_template
get_journey
get_journey_date_range_kpi
get_journey_execution_activity_metrics
get_journey_execution_metrics
get_journey_run_execution_activity_metrics
get_journey_run_execution_metrics
get_journey_runs
get_push_template
get_recommendation_configuration
get_recommendation_configurations
get_segment
get_segment_export_jobs
get_segment_import_jobs
get_segments
get_segment_version
get_segment_versions
get_sms_channel
get_sms_template
get_user_endpoints
get_voice_channel
get_voice_template
list_journeys
list_tags_for_resource
list_templates
list_template_versions
phone_number_validate
put_events
put_event_stream
remove_attributes
send_messages
send_otp_message
send_users_messages
tag_resource
untag_resource
update_adm_channel
update_apns_channel
update_apns_sandbox_channel
update_apns_voip_channel
update_apns_voip_sandbox_channel
update_apns_channel
update_application_settings
update_baidu_channel

Retrieves information about the status and settings of a specific export job for an application.
Retrieves information about the status and settings of all the export jobs for an application.
Retrieves information about the status and settings of the GCM channel for an application.
Retrieves information about the status and settings of a specific import job for an application.
Retrieves information about the status and settings of all the import jobs for an application.
Retrieves the in-app messages targeted for the provided endpoint ID.
Retrieves the content and settings of a message template for messages sent through the in-app channel.
Retrieves information about the status, configuration, and other settings for a journey.
Retrieves (queries) pre-aggregated data for a standard engagement metric that applies to journeys.
Retrieves (queries) pre-aggregated data for a standard execution metric that applies to journeys.
Retrieves (queries) pre-aggregated data for a standard run execution metric that applies to journeys.
Provides information about the runs of a journey.
Retrieves the content and settings of a message template for messages that are sent through the SMS channel.
Retrieves information about an Amazon Pinpoint configuration for a recommender model.
Retrieves information about all the recommender model configurations that are associated with an application.
Retrieves information about the status and settings of the export jobs for a segment.
Retrieves information about the status and settings of the import jobs for a segment.
Retrieves information about the configuration, dimension, and other settings for the SMS channel for an application.
Retrieves the content and settings of a message template for messages that are sent through the voice channel.
Retrieves information about all the endpoints that are associated with a specific segment.
Retrieves information about the status and settings of the voice channel for an application.
Retrieves the content and settings of a message template for messages that are sent through a push notification channel.
Retrieves information about all the tags (keys and values) that are associated with an application.
Retrieves information about all the message templates that are associated with a specific segment.
Retrieves information about all the versions of a specific message template.
Retrieves information about a phone number.
Creates a new event to record for endpoints, or creates or updates endpoint data.
Creates a new event stream for an application or updates the settings of an existing event stream.
Removes one or more custom attributes, of the same attribute type, from the application.
Creates and sends a direct message.
Send an OTP message.
Creates and sends a message to a list of users.
Adds one or more tags (keys and values) to an application, campaign, message template, segment, or user.
Removes one or more tags (keys and values) from an application, campaign, message template, segment, or user.
Enables the ADM channel for an application or updates the status and settings for the ADM channel.
Enables the APNs channel for an application or updates the status and settings for the APNs channel.
Enables the APNs sandbox channel for an application or updates the status and settings for the APNs sandbox channel.
Enables the APNs VoIP channel for an application or updates the status and settings for the APNs VoIP channel.
Enables the APNs VoIP sandbox channel for an application or updates the status and settings for the APNs VoIP sandbox channel.
Enables the Baidu channel for an application or updates the status and settings for the Baidu channel.
Amazon Pinpoint Email Service

Description

Welcome to the Amazon Pinpoint Email API Reference. This guide provides information about the Amazon Pinpoint Email API (version 1.0), including supported operations, data types, parameters, and schemas.

Amazon Pinpoint is an AWS service that you can use to engage with your customers across multiple messaging channels. You can use Amazon Pinpoint to send email, SMS text messages, voice...
messages, and push notifications. The Amazon Pinpoint Email API provides programmatic access to options that are unique to the email channel and supplement the options provided by the Amazon Pinpoint API.

If you’re new to Amazon Pinpoint, you might find it helpful to also review the Amazon Pinpoint Developer Guide. The Amazon Pinpoint Developer Guide provides tutorials, code samples, and procedures that demonstrate how to use Amazon Pinpoint features programmatically and how to integrate Amazon Pinpoint functionality into mobile apps and other types of applications. The guide also provides information about key topics such as Amazon Pinpoint integration with other AWS services and the limits that apply to using the service.

The Amazon Pinpoint Email API is available in several AWS Regions and it provides an endpoint for each of these Regions. For a list of all the Regions and endpoints where the API is currently available, see AWS Service Endpoints in the Amazon Web Services General Reference. To learn more about AWS Regions, see Managing AWS Regions in the Amazon Web Services General Reference.

In each Region, AWS maintains multiple Availability Zones. These Availability Zones are physically isolated from each other, but are united by private, low-latency, high-throughput, and highly redundant network connections. These Availability Zones enable us to provide very high levels of availability and redundancy, while also minimizing latency. To learn more about the number of Availability Zones that are available in each Region, see AWS Global Infrastructure.

Usage

```
pinpointemail(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

Arguments

configure Optional configuration of credentials, endpoint, and/or region.

• credentials:
  • creds:
    * access_key_id: AWS access key ID
    * secret_access_key: AWS secret access key
    * session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

• endpoint: The complete URL to use for the constructed client.
• region: The AWS Region used in instantiating the client.
• close_connection: Immediately close all HTTP connections.
• timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.

- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [URL](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

`credentials` Optional credentials shorthand for the `config` parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.

- **anonymous**: Set anonymous credentials.

`endpoint` Optional shorthand for complete URL to use for the constructed client.

`region` Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- pinpointemail(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    )
  )
)
```
pinpointemail

),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

create_configuration_set  Create a configuration set
create_configuration_set_event_destination  Create an event destination
create_dedicated_ip_pool  Create a new pool of dedicated IP addresses
create_deliverability_test_report  Create a new predictive inbox placement test
create_email_identity  Verifies an email identity for use with Amazon Pinpoint
delete_configuration_set  Delete an existing configuration set
delete_configuration_set_event_destination  Delete an event destination
delete_dedicated_ip_pool  Delete a dedicated IP pool
delete_email_identity  Deletes an email identity that you previously verified for use with Amazon Pinpoint
get_account  Obtain information about the email-sending status and capabilities of your Amazon Pinpoint account
get_blacklist_reports  Retrieve a list of the blacklists that your dedicated IP addresses appear on
get_configuration_set  Get information about an existing configuration set, including the dedicated IP pool that it's associated with, whether or not it's enabled for sending email, and more
get_configuration_set_event_destinations  Retrieve a list of event destinations that are associated with a configuration set
get_dedicated_ip  Get information about a dedicated IP address, including the name of the dedicated IP pool that it's associated with, as well information about the automatic warm-up process for the address
get_dedicated_ips  List the dedicated IP addresses that are associated with your Amazon Pinpoint account
get_dedicated_ips  Get information about a dedicated IP address, including the name of the dedicated IP pool that it's associated with, as well information about the automatic warm-up process for the address
get_delivery_action_support  Get information about whether Amazon Pinpoint delivers to bulk email service providers
get_domain_deliverability_campaign  Retrieve deliverability data for all the campaigns that used a specific domain to send email during a specified time range
get_domain_statistics_report  Retrieve inbox placement and engagement rates for the domains that you use to send email
get_email_identity  Provides information about a specific identity associated with your Amazon Pinpoint account, including the identity's verification status, its DKIM authentication status, and its custom Mail-From settings
list_configuration_sets  List all of the configuration sets associated with your Amazon Pinpoint account in the current AWS Region
list_dedicated_ip_pools  List all of the dedicated IP pools that exist in your Amazon Pinpoint account
list_deliverability_test_reports  Show a list of the predictive inbox placement tests that you've performed, regardless of their statuses
list_domain_deliverability_campaigns  List all of the email identities that are associated with your Amazon Pinpoint account
list_email_identities  List all the tags (keys and values) that are associated with a specified resource
put_account_dedicated_ip_warmup_attributes  Enable or disable the automatic warm-up feature for dedicated IP addresses
put_account_sending_attributes  Enable or disable the ability of your account to send email
put_configuration_set_delivery_options  Associate a configuration set with a dedicated IP pool
put_configuration_set_event_destination  Enable or disable collection of reputation metrics for emails that you send using a particular configuration set in a specific AWS Region
put_configuration_set_reputation_options  Specify a custom domain to use for open and click tracking elements in email
put_configuration_set_sending_options  Move a custom domain to use for open and click tracking elements in email
put_configuration_set_tracking_options  Put dedicated ip warmup attributes
put_dedicated_ip_in_pool  Enable or disable the Deliverability dashboard for your Amazon Pinpoint account
put_dedicated_ip_warmup_attributes  Used to enable or disable DKIM authentication for an email identity
put_email_identity_dkim_attributes  Used to enable or disable feedback forwarding for an identity
put_email_identity_feedforward_attributes  Used to enable or disable the custom Mail-From domain configuration for an identity
Example:
```r
## Not run:
svc <- pinpointemail()
svc$create_configuration_set(
  Foo = 123
)

## End(Not run)
```

**Description**

Pinpoint SMS and Voice Messaging public facing APIs

**Usage**

```r
credentials = list(),
edpoint = NULL,
region = NULL
)
```

**Arguments**

- `config`: Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
    - `creds`:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
    - `anonymous`: Set anonymous credentials.
  - `endpoint`: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoint.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoint.html)

**credentials** Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
cvc <- pinpointsmsvoice(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
```
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

create_configuration_set Create a new configuration set
create_configuration_set_event_destination Create a new event destination in a configuration set
delete_configuration_set Deletes an existing configuration set
delete_configuration_set_event_destination Deletes an event destination in a configuration set
get_configuration_set_event_destinations Obtain information about an event destination, including the types of events it reports, the Amazon Resource Name (ARN) of the destination, and the name of the event destination
list_configuration_sets List all of the configuration sets associated with your Amazon Pinpoint account
send_voice_message Create a new voice message and send it to a recipient’s phone number
update_configuration_set_event_destination Update an event destination in a configuration set

Examples

```r
## Not run:
svc <- pinpointsmsvoice()
svc$create_configuration_set(
    Foo = 123
)

## End(Not run)
```

Description

Welcome to the Amazon Pinpoint SMS and Voice, version 2 API Reference. This guide provides information about Amazon Pinpoint SMS and Voice, version 2 API resources, including supported HTTP methods, parameters, and schemas.

Amazon Pinpoint is an Amazon Web Services service that you can use to engage with your recipients across multiple messaging channels. The Amazon Pinpoint SMS and Voice, version 2 API
provides programmatic access to options that are unique to the SMS and voice channels. Amazon Pinpoint SMS and Voice, version 2 resources such as phone numbers, sender IDs, and opt-out lists can be used by the Amazon Pinpoint API.

If you’re new to Amazon Pinpoint SMS, it’s also helpful to review the Amazon Pinpoint SMS User Guide. The Amazon Pinpoint Developer Guide provides tutorials, code samples, and procedures that demonstrate how to use Amazon Pinpoint SMS features programmatically and how to integrate Amazon Pinpoint functionality into mobile apps and other types of applications. The guide also provides key information, such as Amazon Pinpoint integration with other Amazon Web Services services, and the quotas that apply to use of the service.

Regional availability

The Amazon Pinpoint SMS and Voice, version 2 API Reference is available in several Amazon Web Services Regions and it provides an endpoint for each of these Regions. For a list of all the Regions and endpoints where the API is currently available, see Amazon Web Services Service Endpoints and Amazon Pinpoint endpoints and quotas in the Amazon Web Services General Reference. To learn more about Amazon Web Services Regions, see Managing Amazon Web Services Regions in the Amazon Web Services General Reference.

In each Region, Amazon Web Services maintains multiple Availability Zones. These Availability Zones are physically isolated from each other, but are united by private, low-latency, high-throughput, and highly redundant network connections. These Availability Zones enable us to provide very high levels of availability and redundancy, while also minimizing latency. To learn more about the number of Availability Zones that are available in each Region, see Amazon Web Services Global Infrastructure.

Usage

```r
pinpointsmsvoicev2(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**: 
    - **creds**: 
      * **access_key_id**: AWS access key ID
      * **secret_access_key**: AWS secret access key
      * **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html

credentials: Optional credentials shorthand for the config parameter

- creds:
  - access_key_id: AWS access key ID
  - secret_access_key: AWS secret access key
  - session_token: AWS temporary session token

- profile: The name of a profile to use. If not given, then the default profile is used.

- anonymous: Set anonymous credentials.

documents

endpoint: Optional shorthand for complete URL to use for the constructed client.

region: Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
cvc <- pinpointsmsvoicev2(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string",
  close_connection = "logical",
  timeout = "numeric",
  s3_force_path_style = "logical",
  sts_regional_endpoint = "string"
)
```
secret_access_key = "string",
session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

associate_origination_identity Associates the specified origination identity with a pool
associate_protect_configuration Associate a protect configuration with a configuration set
create_configuration_set Creates a new configuration set
create_event_destination Creates a new event destination in a configuration set
create_opt_out_list Creates a new opt-out list
create_pool Creates a new pool and associates the specified origination identity to the pool
create_protect_configuration Create a new protect configuration
create_registration Associate the registration with an origination identity such as a phone number or sender ID
create_registration_attachment Create a new registration attachment to use for uploading a file or a URL to a file
create_registration_version Create a new version of the registration and increase the VersionNumber
create_verified_destination_number You can only send messages to verified destination numbers when your account is in the sandbox
delete_account_default_protect_configuration Removes the current account default protect configuration
delete_configuration_set Deletes an existing configuration set
delete_default_message_type Deletes an existing default message type on a configuration set
delete_default_sender_id Deletes an existing default sender ID on a configuration set
delete_event_destination Deletes an existing event destination
delete_keyword Deletes an existing keyword from an origination phone number or pool
delete_media_message_spend_limit_override Deletes an account-level monthly spending limit override for sending multimedia messages (MMS)
delete_opted_out_number Deletes an existing opted out destination phone number from the specified opt-out list
delete_opt_out_list Deletes an existing opt-out list
delete_pool Deletes an existing pool
delete_protect_configuration Permanently delete the protect configuration
delete_registration Permanently delete an existing registration from your account
delete_registration_attachment Permanently delete the specified registration attachment
delete_registration_field_value Delete the value in a registration form field
delete_text_message_spend_limit_override Deletes an account-level monthly spending limit override for sending text messages
delete_verified_destination_number Delete a verified destination phone number
delete_voice_message_spend_limit_override Deletes an account level monthly spend limit override for sending voice messages
describe_account_attributes Describes attributes of your Amazon Web Services account
describe_account_limits Describes the current Amazon Pinpoint SMS Voice V2 resource quotas for your account
describe_configuration_sets Describes the specified configuration sets or all in your account
describe_keywords Describes the specified keywords or all keywords on your origination phone number or pool
describe_opted_out_numbers Describes the specified opted out destination numbers or all opted out destinations in your account
describe_opt_out_lists Describes the specified opt-out list or all opt-out lists in your account
describe_phone_numbers Describes the specified origination phone number, or all the phone numbers
describe_pools
describe_protect_configurations
describe_registration_attachments
describe_registration_field_definitions
describe_registration_field_values
describeregistrations
describe_registration_section_definitions
describe_registration_type_definitions
describe_registration_versions
describesender_ids
describe_spend_limits
describe_verified_destination_numbers
disassociate_origination_identity
disassociate_protect_configuration
discard_registration_version
get_protect_configuration_country_rule_set
list_pool_origination_identities
list_registration_associations
list_tags_for_resource
put_keyword
put_opted_out_number
put_registration_field_value
release_phone_number
release_sender_id
request_phone_number
request_sender_id
send_destination_number_verification_code
send_media_message
send_text_message
send_voice_message
set_account_default_protect_configuration
set_default_message_type
set_default_sender_id
set_media_message_spend_limit_override
set_text_message_spend_limit_override
set_voice_message_spend_limit_override
submit_registration_version
tag_resource
untag_resource
update_event_destination
update_phone_number
update_pool
update_protect_configuration
update_protect_configuration_country_rule_set
update_sender_id
verify_destination_number

Retrieves the specified pools or all pools associated with your Amazon Web Services account
Retrieves the protect configurations that match any of filters
Retrieves the specified registration attachments or all registration attachments
Retrieves the specified registration type field definitions
Retrieves the specified registration field values
Retrieves the specified registrations
Retrieves the specified registration section definitions
Retrieves the specified registration type definitions
Retrieves the specified registration version
Describes the specified SenderIds or all SenderIds associated with your Amazon Web Services account
Describes the current Amazon Pinpoint monthly spend limits for sending voice and text messages
Retrieves the specified verified destination numbers
Removes the specified origination identity from an existing pool
Disassociate a protect configuration from a configuration set
Discard the current version of the registration
Retrieve the CountryRuleSet for the specified NumberCapability from a protect configuration
Lists all associated origination identities in your pool
Retrieve all of the origination identities that are associated with a registration
List all tags associated with a resource
Creates or updates a keyword configuration on an origination phone number
Creates an opted out destination phone number in the opt-out list
Creates or updates a field value for a registration
Releases an existing origination phone number in your account
Releases an existing sender ID in your account
Request an origination phone number for use in your account
Request a new sender ID that doesn’t require registration
Before you can send test messages to a verified destination phone number you
Create a new multimedia message (MMS) and sends it to a recipient’s phone
Create a new text message and sends it to a recipient’s phone number
Allows you to send a request that sends a voice message through Amazon Pinpoint
Set a protect configuration as your account default
Sets the default message type on a configuration set
Sets default sender ID on a configuration set
Sets an account level monthly spend limit override for sending MMS messages
Sets an account level monthly spend limit override for sending text messages
Sets an account level monthly spend limit override for sending voice messages
Submit the specified registration for review and approval
Adds or overwrites only the specified tags for the specified Amazon Pinpoint resource
Removes the association of the specified tags from an Amazon Pinpoint SMS Voice V2 resource
Updates an existing event destination in a configuration set
Updates the configuration of an existing origination phone number
Updates the configuration of an existing pool
Update the setting for an existing protect configuration
Update a country rule set to ALLOW or BLOCK messages to be sent to the
Updates the configuration of an existing sender ID
Use the verification code that was received by the verified destination phone
Examples

```r
## Not run:
svc <- pinpointsmsvoicev2()
svc$associate_origination_identity(
  Foo = 123
)
## End(Not run)
```

---

### Description

Amazon Polly is a web service that makes it easy to synthesize speech from text.

The Amazon Polly service provides API operations for synthesizing high-quality speech from plain text and Speech Synthesis Markup Language (SSML), along with managing pronunciations lexicons that enable you to get the best results for your application domain.

### Usage

```r
polly(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

### Arguments

- `config`: Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
    - `creds`:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
    - `anonymous`: Set anonymous credentials.
  - `endpoint`: The complete URL to use for the constructed client.
  - `region`: The AWS Region used in instantiating the client.
  - `close_connection`: Immediately close all HTTP connections.
  - `timeout`: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - `s3_force_path_style`: Set this to true to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
credentials  Optional credentials shorthand for the config parameter
  - creds:
    - access_key_id: AWS access key ID
    - secret_access_key: AWS secret access key
    - session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.

endpoint  Optional shorthand for complete URL to use for the constructed client.

region  Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

dsvc <- polly(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
...
region = "string"
)

## Operations

delete_lexicon

describe_voices

get_lexicon

get_speech_synthesis_task

list_lexicons

list_speech_synthesis_tasks

put_lexicon

start_speech_synthesis_task

synthesize_speech

# Deletes the specified pronunciation lexicon stored in an Amazon Web Services Region

# Returns the list of voices that are available for use when requesting speech synthesis

# Retruns the content of the specified pronunciation lexicon stored in an Amazon Web Services Region

# Retrieves a specific SpeechSynthesisTask object based on its TaskID

# Returns a list of pronunciation lexicons stored in an Amazon Web Services Region

# Returns a list of SpeechSynthesisTask objects ordered by their creation date

# Stores a pronunciation lexicon in an Amazon Web Services Region

# Allows the creation of an asynchronous synthesis task, by starting a new SpeechSynthesisTask

# Synthesizes UTF-8 input, plain text or SSML, to a stream of bytes

## Examples

```r
## Not run:
svc <- polly()
# Deletes a specified pronunciation lexicon stored in an AWS Region.
svc$delete_lexicon(
  Name = "example"
)
```

## AWS Price List Service

### Description

The Amazon Web Services Price List API is a centralized and convenient way to programmatically query Amazon Web Services for services, products, and pricing information. The Amazon Web Services Price List uses standardized product attributes such as Location, Storage Class, and Operating System, and provides prices at the SKU level. You can use the Amazon Web Services Price List to do the following:

- Build cost control and scenario planning tools
- Reconcile billing data
- Forecast future spend for budgeting purposes
- Provide cost benefit analysis that compare your internal workloads with Amazon Web Services
Use GetServices without a service code to retrieve the service codes for all Amazon Web Services, then GetServices with a service code to retrieve the attribute names for that service. After you have the service code and attribute names, you can use get_attribute_values to see what values are available for an attribute. With the service code and an attribute name and value, you can use get_products to find specific products that you’re interested in, such as an AmazonEC2 instance, with a Provisioned IOPS volumeType.

For more information, see Using the Amazon Web Services Price List API in the Billing User Guide.

Usage

pricing(config = list(), credentials = list(), endpoint = NULL, region = NULL)

Arguments

config Optional configuration of credentials, endpoint, and/or region.
• credentials:
  – creds:
    * access_key_id: AWS access key ID
    * secret_access_key: AWS secret access key
    * session_token: AWS temporary session token
  – profile: The name of a profile to use. If not given, then the default profile is used.
  – anonymous: Set anonymous credentials.
• endpoint: The complete URL to use for the constructed client.
• region: The AWS Region used in instantiating the client.
• close_connection: Immediately close all HTTP connections.
• timeout: The time in seconds till a timeout exception is thrown when attempted to make a connection. The default is 60 seconds.
• s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter
• creds:
  – access_key_id: AWS access key ID
  – secret_access_key: AWS secret access key
  – session_token: AWS temporary session token
• profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.
Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- pricing(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- **describe_services** Returns the metadata for one service or a list of the metadata for all services
- **get_attribute_values** Returns a list of attribute values
- **get_price_list_file_url** This feature is in preview release and is subject to change
- **get_products** Returns a list of all products that match the filter criteria
- **list_price_lists** This feature is in preview release and is subject to change
prometheusservice

Examples

```r
## Not run:
svc <- pricing()
svc$describe_services(
  Foo = 123
)
## End(Not run)
```

prometheusservice  Amazon Prometheus Service

Description

Amazon Managed Service for Prometheus is a serverless, Prometheus-compatible monitoring service for container metrics that makes it easier to securely monitor container environments at scale. With Amazon Managed Service for Prometheus, you can use the same open-source Prometheus data model and query language that you use today to monitor the performance of your containerized workloads, and also enjoy improved scalability, availability, and security without having to manage the underlying infrastructure.

For more information about Amazon Managed Service for Prometheus, see the Amazon Managed Service for Prometheus User Guide.

Amazon Managed Service for Prometheus includes two APIs.

- Use the Amazon Web Services API described in this guide to manage Amazon Managed Service for Prometheus resources, such as workspaces, rule groups, and alert managers.
- Use the Prometheus-compatible API to work within your Prometheus workspace.

Usage

```r
prometheusservice(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.
  
- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy `https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html`

**credentials** Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- prometheusservice(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
```

### Notes

- **prometheusservice**: A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.
prometheusservice

```python
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
  creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
  ),
  profile = "string",
  anonymous = "logical"
),
endpoint = "string",
region = "string"
)
```

Operations

- `create_alert_manager_definition`: The CreateAlertManagerDefinition operation creates the alert manager definition in a workspace.
- `create_logging_configuration`: The CreateLoggingConfiguration operation creates a logging configuration for the workspace.
- `create_rule_groups_namespace`: The CreateRuleGroupsNamespace operation creates a rule groups namespace within a workspace.
- `create_scraper`: The CreateScraper operation creates a scraper to collect metrics.
- `create_workspace`: Creates a Prometheus workspace.
- `delete_alert_manager_definition`: Deletes the alert manager definition from a workspace.
- `delete_logging_configuration`: Deletes the logging configuration for a workspace.
- `delete_rule_groups_namespace`: Deletes one rule groups namespace and its associated rule groups definition.
- `delete_scraper`: The DeleteScraper operation deletes one scraper, and stops any metrics collection that the scraper is performing.
- `delete_workspace`: Deletes an existing workspace.
- `describe_alert_manager_definition`: Retrieves the full information about the alert manager definition for a workspace.
- `describe_logging_configuration`: Returns complete information about the current logging configuration of the workspace.
- `describe_rule_groups_namespace`: Returns complete information about one rule groups namespace.
- `describe_scraper`: The DescribeScraper operation displays information about an existing scraper.
- `describe_workspace`: Returns information about an existing workspace.
- `get_default_scraper_configuration`: The GetDefaultScraperConfiguration operation returns the default scraper configuration used when Amazon EKS creates a scraper for you.
- `list_rule_groups_namespaces`: Lists all of the rule groups namespaces in a workspace.
- `list_scrapers`: The ListScrapers operation lists all of the scrapers in your account.
- `list_tags_for_resource`: Lists all of the Amazon Managed Service for Prometheus workspaces in your account.
- `list_workspaces`: Updates an existing alert manager definition in a workspace.
- `put_alert_manager_definition`: Updates an existing rule groups namespace within a workspace.
- `put_rule_groups_namespace`: The TagResource operation associates tags with an Amazon Managed Service for Prometheus workspace.
- `tag_resource`: Removes the specified tags from an Amazon Managed Service for Prometheus resource.
- `untag_resource`: Updates the log group ARN or the workspace ID of the current logging configuration.
- `update_logging_configuration`: Updates the alias of an existing workspace.
Examples

```r
## Not run:
svc <- prometheusservice()
svc$create_alert_manager_definition(
  Foo = 123
)
## End(Not run)
```

---

### Description

This is the Proton Service API Reference. It provides descriptions, syntax and usage examples for each of the actions and data types for the Proton service.

The documentation for each action shows the Query API request parameters and the XML response. Alternatively, you can use the Amazon Web Services CLI to access an API. For more information, see the Amazon Web Services Command Line Interface User Guide.

The Proton service is a two-pronged automation framework. Administrators create service templates to provide standardized infrastructure and deployment tooling for serverless and container based applications. Developers, in turn, select from the available service templates to automate their application or service deployments.

Because administrators define the infrastructure and tooling that Proton deploys and manages, they need permissions to use all of the listed API operations.

When developers select a specific infrastructure and tooling set, Proton deploys their applications. To monitor their applications that are running on Proton, developers need permissions to the service `create`, `list`, `update` and `delete` API operations and the service instance `list` and `update` API operations.

To learn more about Proton, see the Proton User Guide.

### Ensuring Idempotency

When you make a mutating API request, the request typically returns a result before the asynchronous workflows of the operation are complete. Operations might also time out or encounter other server issues before they're complete, even if the request already returned a result. This might make it difficult to determine whether the request succeeded. Moreover, you might need to retry the request multiple times to ensure that the operation completes successfully. However, if the original request and the subsequent retries are successful, the operation occurs multiple times. This means that you might create more resources than you intended.

Idempotency ensures that an API request action completes no more than one time. With an idempotent request, if the original request action completes successfully, any subsequent retries complete successfully without performing any further actions. However, the result might contain updated information, such as the current creation status.

The following lists of APIs are grouped according to methods that ensure idempotency.
Idempotent create APIs with a client token

The API actions in this list support idempotency with the use of a client token. The corresponding Amazon Web Services CLI commands also support idempotency using a client token. A client token is a unique, case-sensitive string of up to 64 ASCII characters. To make an idempotent API request using one of these actions, specify a client token in the request. We recommend that you don’t reuse the same client token for other API requests. If you don’t provide a client token for these APIs, a default client token is automatically provided by SDKs.

Given a request action that has succeeded:

If you retry the request using the same client token and the same parameters, the retry succeeds without performing any further actions other than returning the original resource detail data in the response.

If you retry the request using the same client token, but one or more of the parameters are different, the retry throws a ValidationException with an IdempotentParameterMismatch error.

Client tokens expire eight hours after a request is made. If you retry the request with the expired token, a new resource is created.

If the original resource is deleted and you retry the request, a new resource is created.

Idempotent create APIs with a client token:

- CreateEnvironmentTemplateVersion
- CreateServiceTemplateVersion
- CreateEnvironmentAccountConnection

Idempotent create APIs

Given a request action that has succeeded:

If you retry the request with an API from this group, and the original resource hasn’t been modified, the retry succeeds without performing any further actions other than returning the original resource detail data in the response.

If the original resource has been modified, the retry throws a ConflictException.

If you retry with different input parameters, the retry throws a ValidationException with an IdempotentParameterMismatch error.

Idempotent create APIs:

- CreateEnvironmentTemplate
- CreateServiceTemplate
- CreateEnvironment
- CreateService

Idempotent delete APIs

Given a request action that has succeeded:

When you retry the request with an API from this group and the resource was deleted, its metadata is returned in the response.

If you retry and the resource doesn’t exist, the response is empty.

In both cases, the retry succeeds.

Idempotent delete APIs:
• DeleteEnvironmentTemplate
• DeleteEnvironmentTemplateVersion
• DeleteServiceTemplate
• DeleteServiceTemplateVersion
• DeleteEnvironmentAccountConnection

**Asynchronous idempotent delete APIs**

Given a request action that has succeeded:

If you retry the request with an API from this group, if the original request delete operation status is DELETE_IN_PROGRESS, the retry returns the resource detail data in the response without performing any further actions.

If the original request delete operation is complete, a retry returns an empty response.

Asynchronous idempotent delete APIs:

• DeleteEnvironment
• DeleteService

**Usage**

proton(config = list(), credentials = list(), endpoint = NULL, region = NULL)

**Arguments**

config Optional configuration of credentials, endpoint, and/or region.

• credentials:
  – creds:
    * access_key_id: AWS access key ID
    * secret_access_key: AWS secret access key
    * session_token: AWS temporary session token
  – profile: The name of a profile to use. If not given, then the default profile is used.
  – anonymous: Set anonymous credentials.

• endpoint: The complete URL to use for the constructed client.

• region: The AWS Region used in instantiating the client.

• close_connection: Immediately close all HTTP connections.

• timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

• s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

• sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter

• creds:
proton

- **access_key_id**: AWS access key ID
- **secret_access_key**: AWS secret access key
- **session_token**: AWS temporary session token

  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

**endpoint**

Optional shorthand for complete URL to use for the constructed client.

**region**

Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- proton(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```
Operations

- In a management account, an environment account connection request is accepted.
- Attempts to cancel a component deployment (for a component that is in the IN_PROGRESS deployment status).
- Attempts to cancel an environment deployment on an UpdateEnvironment action.
- Attempts to cancel a service instance deployment on an UpdateServiceInstance action.
- Attempts to cancel a service pipeline deployment on an UpdateServicePipeline action.
- Create a Proton component
- Deploy a new environment
- Create an environment account connection in an environment account so that environment infrastructure resources can be provisioned in the environment account from a management account.
- Create a new major or minor version of an environment template
- Create and register a link to a repository
- Create a Proton service
- Create a service instance
- Create the Proton Ops configuration file
- Create a service template
- Create a new major or minor version of a service template
- Set up a template to create new template versions automatically by tracking a linked repository.
- Delete a Proton component resource
- Delete the deployment
- Delete an environment
- In an environment account, delete an environment account connection
- If no other major or minor versions of an environment template exist, delete the template.
- If no other minor versions of an environment template exist, delete a major version of the template.
- De-register and unlink your repository
- Delete a service, with its instances and pipeline
- Delete the Proton Ops file
- If no other major or minor versions of the service template exist, delete the service template.
- If no other minor versions of a service template exist, delete a major version of the service template.
- Delete a template sync configuration
- Get detail data for Proton account-wide settings
- Get detailed data for a component
- Get detailed data for a deployment
- Get detailed data for an environment
- In an environment account, get the detailed data for an environment account connection.
- Get detailed data for an environment template
- Get detailed data for a major or minor version of an environment template
- Get detail data for a linked repository
- Get the sync status of a repository used for Proton template sync
- Get counts of Proton resources
- Get detailed data for a service
- Get detailed data for a service instance
- Get the status of the synced service instance
- Get detailed data for the service sync blocker summary
- Get detailed information for the service sync configuration
- Get detailed data for a service template
- Get detailed data for a major or minor version of a service template
proton

get_template_sync_config — Get detail data for a template sync configuration
get_template_sync_status — Get the status of a template sync
list_component_outputs — Get a list of component Infrastructure as Code (IaC) outputs
list_component_provisioned_resources — List provisioned resources for a component with details
list_components — List components with summary data
list_deployments — List deployments
list_environment_account_connections — View a list of environment account connections
list_environment_outputs — List the infrastructure as code outputs for your environment
list_environment_provisioned_resources — List the provisioned resources for your environment
list_environments — List environments with detail data summaries
list_environment_templates — List environments
list_environment_template_versions — List major or minor versions of an environment template with detail data
list_repositories — List linked repositories with detail data
list_repository_sync_definitions — List repository sync definitions with detail data
list_service_instance_outputs — Get a list service of instance Infrastructure as Code (IaC) outputs
list_service_instance_provisioned_resources — List provisioned resources for a service instance with details
list_service_instances — List service instances with summary data
list_service_pipeline_outputs — Get a list of service pipeline Infrastructure as Code (IaC) outputs
list_service_pipeline_provisioned_resources — List services with summaries of detail data
list_service_templates — List services
list_service_template_versions — List major or minor versions of a service template with detail data
list_tags_for_resource — List tags for a resource
notify_resource_deployment_status_change — Notify Proton of status changes to a provisioned resource when you use self-managed provisioning
reject_environment_account_connection — In a management account, reject an environment account connection from another environment account
set_environment_account_connection — In an environment account, update an environment account connection to use a new IAM role
tag_resource — Tag a resource
untag_resource — Remove a customer tag from a resource
update_account_settings — Update Proton settings that are used for multiple services in the Amazon Web Services account
update_component — Update a component
update_environment — Update an environment
update_environment_account_connection — In an environment account, update an environment account connection to use a new IAM role
update_environment_template — Update an environment template
update_environment_template_version — Update a major or minor version of an environment template
update_service — Edit a service description or use a spec to add and delete service instances
update_service_instance — Update a service instance
update_service_pipeline — Update the service pipeline
update_service_sync_blocker — Update the service sync blocker by resolving it
update_service_sync_config — Update the Proton Ops config file
update_service_template — Update a service template
update_service_template_version — Update a major or minor version of a service template
update_template_sync_config — Update template sync configuration parameters, except for the templateName and templateType

Examples

```r
## Not run:
svc <- proton()
svc$accept_environment_account_connection()
```
qldb

Amazon QLDB

Description

The resource management API for Amazon QLDB

Usage

qldb(config = list(), credentials = list(), endpoint = NULL, region = NULL)

Arguments

config

Optional configuration of credentials, endpoint, and/or region.

• credentials:
  – creds:
    * access_key_id: AWS access key ID
    * secret_access_key: AWS secret access key
    * session_token: AWS temporary session token
  – profile: The name of a profile to use. If not given, then the default profile is used.
  – anonymous: Set anonymous credentials.
• endpoint: The complete URL to use for the constructed client.
• region: The AWS Region used in instantiating the client.
• close_connection: Immediately close all HTTP connections.
• timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials

Optional credentials shorthand for the config parameter

• creds:
  – access_key_id: AWS access key ID
  – secret_access_key: AWS secret access key
  – session_token: AWS temporary session token

Foo = 123

## End(Not run)
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**
A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**
```
svc <- qldb(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

**Operations**
- `cancel_journal_kinesis_stream` Ends a given Amazon QLDB journal stream
create_ledger
delete_ledger
describe_journal_kinesis_stream
describe_journal_s3_export
describe_ledger
export_journal_to_s3
get_block
get_digest
get_revision
list_journal_kinesis_streams_for_ledger
list_journal_s3_exports
list_journal_s3_exports_for_ledger
list_ledgers
list_tags_for_resource
stream_journal_to_kinesis
tag_resource
untag_resource
update_ledger
update_ledger_permissions_mode

Creates a new ledger in your Amazon Web Services account in the current Region
Deletes a ledger and all of its contents
Returns detailed information about a given Amazon QLDB journal stream
Returns information about a journal export job, including the ledger name, export ID
Returns information about a ledger, including its state, permissions mode, encryption
Exports journal contents within a date and time range from a ledger into a specified S3
Returns a block object at a specified address in a journal
Returns the digest of a ledger at the latest committed block in the journal
Returns a revision data object for a specified document ID and block address
Returns all Amazon QLDB journal streams for a given ledger
Returns all journal export jobs for all ledgers that are associated with the current AWS account
Returns all journal export jobs for a specified ledger
Returns all ledgers that are associated with the current Amazon Web Services account
Returns all tags for a specified Amazon QLDB resource
Creates a journal stream for a given Amazon QLDB ledger
Adds one or more tags to a specified Amazon QLDB resource
Removes one or more tags from a specified Amazon QLDB resource
Updates properties on a ledger
Updates the permissions mode of a ledger

Examples

```r
## Not run:
svc <- qldb()
svc$cancel_journal_kinesis_stream(
  Foo = 123
)

## End(Not run)
```

qdbsession

Amazon QLDB Session

Description

The transactional data APIs for Amazon QLDB
Instead of interacting directly with this API, we recommend using the QLDB driver or the QLDB shell to execute data transactions on a ledger.

- If you are working with an AWS SDK, use the QLDB driver. The driver provides a high-level abstraction layer above this QLDB Session data plane and manages send_command API calls for you. For information and a list of supported programming languages, see Getting started with the driver in the Amazon QLDB Developer Guide.
- If you are working with the AWS Command Line Interface (AWS CLI), use the QLDB shell. The shell is a command line interface that uses the QLDB driver to interact with a ledger. For information, see Accessing Amazon QLDB using the QLDB shell.
qldbsession

Usage

qldbsession(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

config Optional configuration of credentials, endpoint, and/or region.
  - credentials:
    * access_key_id: AWS access key ID
    * secret_access_key: AWS secret access key
    * session_token: AWS temporary session token
    * profile: The name of a profile to use. If not given, then the default profile is used.
    * anonymous: Set anonymous credentials.
  * endpoint: The complete URL to use for the constructed client.
  * region: The AWS Region used in instantiating the client.
  * close_connection: Immediately close all HTTP connections.
  * timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  * s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

credentials Optional credentials shorthand for the config parameter
  - creds:
    * access_key_id: AWS access key ID
    * secret_access_key: AWS secret access key
    * session_token: AWS temporary session token
  * profile: The name of a profile to use. If not given, then the default profile is used.
  * anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```
svc <- qldbsession(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `send_command` Sends a command to an Amazon QLDB ledger

Examples

```
## Not run:
svc <- qldbsession()
svc$send_command(
  Foo = 123
)

## End(Not run)
```
Amazon QuickSight API Reference

Amazon QuickSight is a fully managed, serverless business intelligence service for the Amazon Web Services Cloud that makes it easy to extend data and insights to every user in your organization. This API reference contains documentation for a programming interface that you can use to manage Amazon QuickSight.

Usage

```python
quicksight(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - **access_key_id**: AWS access key ID
- **secret_access_key**: AWS secret access key
- **session_token**: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- quicksight(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```
quicksight

Operations
cancel_ingestion
create_account_customization
create_account_subscription
create_analysis
create_dashboard
create_data_set
create_data_source
create_folder
create_folder_membership
create_group
create_group_membership
create_iam_policy_assignment
create_ingestion
create_namespace
create_refresh_schedule
create_role_membership
create_template
create_template_alias
create_theme
create_theme_alias
create_topic
create_topic_refresh_schedule
create_vpc_connection
delete_account_customization
delete_account_subscription
delete_analysis
delete_dashboard
delete_data_set
delete_data_set_refresh_properties
delete_data_source
delete_folder
delete_folder_membership
delete_group
delete_group_membership
delete_iam_policy_assignment
delete_identity_propagation_config
delete_namespace
delete_refresh_schedule
delete_role_custom_permission
delete_role_membership
delete_template
delete_template_alias
delete_theme
delete_theme_alias
delete_topic
delete_topic_refresh_schedule
delete_user
delete_user_by_principal_id
Cancels an ongoing ingestion of data into SPICE
Creates Amazon QuickSight customizations for the current Amazon Web Services Region
Creates an Amazon QuickSight account, or subscribes to Amazon QuickSight Q
Creates an analysis in Amazon QuickSight
Creates a dashboard from either a template or directly with a DashboardDefinition
Creates a dataset
Creates a data source
Creates an empty shared folder
Adds an asset, such as a dashboard, analysis, or dataset into a folder
Use the CreateGroup operation to create a group in Amazon QuickSight
Adds an Amazon QuickSight user to an Amazon QuickSight group
Creates an assignment with one specified IAM policy, identified by its Amazon Resource Name (ARN)
Creates and starts a new SPICE ingestion for a dataset
(Enterprise edition only) Creates a new namespace for you to use with Amazon QuickSight
Creates a refresh schedule for a dataset
Use CreateRoleMembership to add an existing Amazon QuickSight group to an existing role
Creates a template either from a TemplateDefinition or from an existing Amazon QuickSight template
Creates a template alias for a template
Creates a theme
Creates a theme alias for a theme
Creates a new Q topic
Creates a topic refresh schedule
Creates a new VPC connection
Deletes all Amazon QuickSight customizations in this Amazon Web Services Region
Use the DeleteAccountSubscription operation to delete an Amazon QuickSight account
Deletes an analysis from Amazon QuickSight
Deletes a dashboard
Deletes a dataset
Deletes the dataset refresh properties of the dataset
Deletes the data source permanently
Deletes an empty folder
Removes an asset, such as a dashboard, analysis, or dataset, from a folder
Removes a user group from Amazon QuickSight
Removes a user from a group so that the user is no longer a member of the group
Deletes an existing IAM policy assignment
Deletes all access scopes and authorized targets that are associated with a service connection
Deletes a namespace and the users and groups that are associated with the namespace
Deletes a refresh schedule from a dataset
Removes custom permissions from the role
Removes a group from a role
Deletes a template
Deletes the item that the specified template alias points to
Deletes a theme
Deletes the version of the theme that the specified theme alias points to
Deletes a topic
Deletes a topic refresh schedule
Deletes the Amazon QuickSight user that is associated with the identity of the IAM user or role
Deletes a user identified by its principal ID
delete_vpc_connection
describe_account_customization
describe_account_settings
describe_account.subscription
describe_analysis
describe_analysis_definition
describe_analysis_permissions
describe_asset_bundle_export_job
describe_asset_bundle_import_job
describe_dashboard
describe_dashboard_definition
describe_dashboard_permissions
describe_dashboard_snapshot_job
describe_dashboard_snapshot_job_result
describe_data_set
describe_data_set_permissions
describe_data_set_refresh_properties
describe_data_source
describe_data_source_permissions
describe_folder
describe_folder_permissions
describe_folder_resolved_permissions
describe_group
describe_group_membership
describe_iam_policy_assignment
describe_ingestion
describe_ip_restriction
describe_namespace
describe_refresh_schedule
describe_role_custom_permission
describe_template
describe_template_alias
describe_template_definition
describe_template_permissions
describe_theme
describe_theme_alias
describe_theme_permissions
describe_topic
describe_topic_permissions
describe_topic_refresh
describe_topic_refresh_schedule
describe_user
describe_vpc_connection
generate_embed_url_for_anonymous_user
generate_embed_url_for_registered_user
get_dashboard_embed_url
get_session_embed_url
list_analyses

Delete a VPC connection
Describes the customizations associated with the provided Amazon Web Services account
Describes the settings that were used when your Amazon QuickSight subscription was first created
Use the DescribeAccountSubscription operation to receive a description of an Amazon QuickSight account
Provides a summary of the metadata for an analysis
Provides a detailed description of the definition of an analysis
Provides the read and write permissions for an analysis
Describes an existing export job
Describes an existing import job
Provides a summary for a dashboard
Provides a detailed description of the definition of a dashboard
Describes read and write permissions for a dashboard
Describes an existing snapshot job
Describes the result of an existing snapshot job that has finished running
Describes a dataset
Describes the permissions on a dataset
Describes the refresh properties of a dataset
Describes a data source
Describes the resource permissions for a data source
Describes a folder
Describes permissions for a folder
Describes the folder resolved permissions
Returns an Amazon QuickSight group’s description and Amazon Resource Name (ARN)
Use the DescribeGroupMembership operation to determine if a user is a member of the specified group
Describes an existing IAM policy assignment, as specified by the assignment name
Describes a SPICE ingestion
Provides a summary and status of IP rules
Describes the current namespace
Provides a summary of a refresh schedule
Describes all custom permissions that are mapped to a role
Describes a template’s metadata
Describes the template alias for a template
Provides a detailed description of the definition of a template
Describes read and write permissions on a template
Describes a theme
Describes the alias for a theme
Describes the read and write permissions for a theme
Describes a topic
Describes the permissions of a topic
Describes the status of a topic refresh
Deletes a topic refresh schedule
Returns information about a user, given the user name
Describes a VPC connection
Generates an embed URL that you can use to embed an Amazon QuickSight dashboard
Generates an embed URL that you can use to embed an Amazon QuickSight experience
Generates a temporary session URL and authorization code(bearer token) that you can use
Generates a session URL and authorization code that you can use to embed the Amazon QuickSight console
Lists Amazon QuickSight analyses that exist in the specified Amazon Web Services account
list_asset_bundle_export_jobs
list_asset_bundle_import_jobs
list_dashboards
list_dashboard_versions
list_data_sets
list_data_sources
list_folder_members
list_folders
list_group_memberships
list_groups
list_iam_policy_assignments
list_iam_policy_assignments_for_user
list_identity_propagation_configs
list_ingestions
list_namespaces
list_refresh_schedules
list_role_memberships
list_tags_for_resource
list_template_aliases
list_templates
list_template_versions
list_theme_aliases
list_themes
list_theme_versions
list_topic_refresh_schedules
list_topics
list_user_groups
list_users
list_vpc_connections
put_data_set_refresh_properties
register_user
restore_analysis
search_analyses
search_dashboards
search_data_sets
search_data_sources
search_folders
search_groups
start_asset_bundle_export_job
start_asset_bundle_import_job
start_dashboard_snapshot_job
tag_resource
untag_resource
update_account_customization
update_account_settings
update_analysis
update_analysis_permissions
update_dashboard

Lists all asset bundle export jobs that have been taken place in the last 14 days
Lists all asset bundle import jobs that have taken place in the last 14 days
Lists dashboards in an Amazon Web Services account
Lists all the versions of the dashboards in the Amazon QuickSight subscription
Lists all of the datasets belonging to the current Amazon Web Services account
Lists data sources in current Amazon Web Services Region that belong to this Amazon Web Services account
List all assets (DASHBOARD, ANALYSIS, and DATASET) in a folder
Lists all folders in an account
Lists member users in a group
Lists all user groups in Amazon QuickSight
Lists the IAM policy assignments in the current Amazon QuickSight account
Lists all of the IAM policy assignments, including the Amazon Resource Names (ARNs) of the services and authorized targets that the Amazon QuickSight IAM Identity Center application can access
Lists the history of SPICE ingestions for a dataset
Lists the namespaces for the specified Amazon Web Services account
Lists the refresh schedules of a dataset
Lists all groups that are associated with a role
Lists the tags assigned to a resource
Lists all the aliases of a template
Lists all the templates in the current Amazon QuickSight account
Lists all the versions of the templates in the current Amazon QuickSight account
Lists all the aliases of a theme
Lists all the themes in the current Amazon Web Services account
Lists all the versions of the themes in the current Amazon Web Services account
Lists all of the refresh schedules for a topic
Lists all of the topics within an account
Lists the Amazon QuickSight groups that an Amazon QuickSight user is a member of
Returns a list of all of the Amazon QuickSight users belonging to this account
Lists all of the VPC connections in the current set Amazon Web Services Region
Creates or updates the dataset refresh properties for the dataset
Creates an Amazon QuickSight user whose identity is associated with the Identity and Access Management (IAM) identity or role specified in the request
Restores an analysis
Searches for analyses that belong to the user specified in the filter
Searches for dashboards that belong to a user
Use the SearchDataSets operation to search for datasets that belong to an account
Use the SearchDataSources operation to search for data sources that belong to an account
Searches the subfolders in a folder
Use the SearchGroups operation to search groups in a specified Amazon QuickSight Region
Starts an Asset Bundle export job
Starts an Asset Bundle import job
Starts an asynchronous job that generates a snapshot of a dashboard’s output
Assigns one or more tags (key-value pairs) to the specified Amazon QuickSight resource
Removes a tag or tags from a resource
Updates Amazon QuickSight customizations for the current Amazon Web Services Region
Updates the Amazon QuickSight settings in your Amazon Web Services account
Updates an analysis in Amazon QuickSight
Updates the read and write permissions for an analysis
Updates a dashboard in an Amazon Web Services account
update_dashboard_links Updates the linked analyses on a dashboard
update_dashboard_permissions Updates read and write permissions on a dashboard
update_data_set_updates the published version of a dashboard
update_data_source Updates a dataset
update_data_source_permissions Updates the permissions on a dataset
update_folder Updates a data source
update_folder_permissions Updates the permissions to a data source
update_group Updates the name of a folder
update_identity_propagation_config Updates permissions of a folder
update_iam_policy_assignment Changes a group description
update_ip_restriction Adds an existing IAM policy assignment
update_ip_restriction Updates the content and status of IP rules
update_public_sharing_settings Use the UpdatePublicSharingSettings operation to turn on or turn off the public sharing settings of a dataset
update_role Custom Permissions Updates the custom permissions that are associated with a role
update_spice_capacity_configuration Updates the SPICE capacity configuration for an Amazon QuickSight account
update_spice_capacity_configuration Updates the template alias of a template
update_template_updates the resource permissions for a template
update_theme Updates the resource permissions for a theme
update_theme_alias Updates a theme
update_theme_permissions Updates the alias of a theme
update_topic Updates the permissions of a topic
update_topic_refresh_schedule Updates a topic
update_user Updates an Amazon QuickSight user
update_vpc_connection Updates a VPC connection

Examples

```r
## Not run:
svc <- quicksight()
svc$cancel_ingestion(
  Foo = 123
)
```

## End(Not run)
Description

This is the Resource Access Manager API Reference. This documentation provides descriptions and syntax for each of the actions and data types in RAM. RAM is a service that helps you securely share your Amazon Web Services resources to other Amazon Web Services accounts. If you use Organizations to manage your accounts, then you can share your resources with your entire organization or to organizational units (OUs). For supported resource types, you can also share resources with individual Identity and Access Management (IAM) roles and users.

To learn more about RAM, see the following resources:

- Resource Access Manager product page
- Resource Access Manager User Guide

Usage

ram(config = list(), credentials = list(), endpoint = NULL, region = NULL)

Arguments

config Optional configuration of credentials, endpoint, and/or region.

- credentials:
  - creds:
    * access_key_id: AWS access key ID
    * secret_access_key: AWS secret access key
    * session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.

- endpoint: The complete URL to use for the constructed client.

- region: The AWS Region used in instantiating the client.

- close_connection: Immediately close all HTTP connections.

- timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

- s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

- sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoint.html

credentials Optional credentials shorthand for the config parameter

- creds:
  - access_key_id: AWS access key ID
  - secret_access_key: AWS secret access key
  - session_token: AWS temporary session token

- profile: The name of a profile to use. If not given, then the default profile is used.

- anonymous: Set anonymous credentials.
endpoint  Optional shorthand for complete URL to use for the constructed client.
region    Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...),
where svc is the name you’ve assigned to the client. The available operations are listed in the Opera-
tions section.

Service syntax

svc <- ram(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)

Operations

accept_resource_share_invitation  Accepts an invitation to a resource share from another Amazon Web Services account
associate_resource_share  Adds the specified list of principals and list of resources to a resource share
associate_resource_share_permission  Adds or replaces the RAM permission for a resource type included in a resource share
create_permission  Creates a customer managed permission for a specified resource type that you
create_permission_version
create_resource_share
delete_permission
delete_permission_version
delete_resource_share
disable_resource_share_permission
enable_resource_share_permission
get_permission
get_resource_policies
group_resource_share_associations
group_resource_share_invitations
get_resource_shares
list_pending_invitation_resources
list_permission_associations
list_permissions
list_permission_versions
list_principals
list_replace_permission_associations_work
list_resources
list_resource_shares
list_resource_share_permissions
list_resource_types
promote_permission_created_from_policy
promote_resource_share_created_from_policy
reject_resource_share_invitation
replace_permission_associations
set_default_permission_version
tag_resource
untag_resource
update_resource_share

Examples

## Not run:
svc <- ram()
svc$accept_resource_share_invitation(
  Foo = 123
)

## End(Not run)
Description

Amazon Relational Database Service (Amazon RDS) is a web service that makes it easier to set up, operate, and scale a relational database in the cloud. It provides cost-efficient, resizeable capacity for an industry-standard relational database and manages common database administration tasks, freeing up developers to focus on what makes their applications and businesses unique.

Amazon RDS gives you access to the capabilities of a MySQL, MariaDB, PostgreSQL, Microsoft SQL Server, Oracle, Db2, or Amazon Aurora database server. These capabilities mean that the code, applications, and tools you already use today with your existing databases work with Amazon RDS without modification. Amazon RDS automatically backs up your database and maintains the database software that powers your DB instance. Amazon RDS is flexible: you can scale your DB instance’s compute resources and storage capacity to meet your application’s demand. As with all Amazon Web Services, there are no up-front investments, and you pay only for the resources you use.

This interface reference for Amazon RDS contains documentation for a programming or command line interface you can use to manage Amazon RDS. Amazon RDS is asynchronous, which means that some interfaces might require techniques such as polling or callback functions to determine when a command has been applied. In this reference, the parameter descriptions indicate whether a command is applied immediately, on the next instance reboot, or during the maintenance window. The reference structure is as follows, and we list following some related topics from the user guide.

Amazon RDS API Reference

- For the alphabetical list of API actions, see API Actions.
- For the alphabetical list of data types, see Data Types.
- For a list of common query parameters, see Common Parameters.
- For descriptions of the error codes, see Common Errors.

Amazon RDS User Guide

- For a summary of the Amazon RDS interfaces, see Available RDS Interfaces.
- For more information about how to use the Query API, see Using the Query API.

Usage

```r
rds(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

- config: Optional configuration of credentials, endpoint, and/or region.
  - credentials:
    - creds:
      - access_key_id: AWS access key ID
      - secret_access_key: AWS secret access key
      - session_token: AWS temporary session token
    - profile: The name of a profile to use. If not given, then the default profile is used.
    - anonymous: Set anonymous credentials.
• **endpoint**: The complete URL to use for the constructed client.
• **region**: The AWS Region used in instantiating the client.
• **close_connection**: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

**credentials**

Optional credentials shorthand for the config parameter

• **creds**:
  – **access_key_id**: AWS access key ID
  – **secret_access_key**: AWS secret access key
  – **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.
• **anonymous**: Set anonymous credentials.

**endpoint**

Optional shorthand for complete URL to use for the constructed client.

**region**

Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- rds(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  )
)
```
rds

),
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

add_role_to_db_cluster
add_role_to_db_instance
add_source_identifier_to_subscription
add_tags_to_resource
apply_pending_maintenance_action
authorize_db_security_group_ingress
backtrack_db_cluster
build_auth_token
cancel_export_task
copy_db_cluster_parameter_group
copy_db_cluster_snapshot
copy_db_parameter_group
copy_db_snapshot
copy_option_group
create_blue_green_deployment
create_custom_db_engine_version
create_db_cluster
create_db_cluster_endpoint
create_db_cluster_parameter_group
create_db_cluster_snapshot
create_db_instance
create_db_instance_read_replica
create_db_parameter_group
create_db_proxy
create_db_proxy_endpoint
create_db_security_group
create_db_shard_group
create_db_snapshot
create_db_subnet_group
create_event_subscription
create_global_cluster
create_integration

Associates an Identity and Access Management (IAM) role with a DB cluster
Associates an Amazon Web Services Identity and Access Management (IAM) role with a DB instance
Adds a source identifier to an existing RDS event notification subscription
Adds metadata tags to an Amazon RDS resource
Applies a pending maintenance action to a resource (for example, to a DB instance)
Enables ingress to a DBSecurityGroup using one of two forms of authorization
Backtracks a DB cluster to a specific time, without creating a new DB cluster
Return an authentication token for a database connection
Cancels an export task in progress that is exporting a snapshot or cluster
Copies the specified DB cluster parameter group
Copies a snapshot of a DB cluster
Copies the specified DB parameter group
Copies the specified DB snapshot
Copies the specified option group
Creates a blue/green deployment
Creates a custom DB engine version (CEV)
Creates a new Amazon Aurora DB cluster or Multi-AZ DB cluster
Creates a new custom endpoint and associates it with an Amazon Aurora DB cluster
Creates a new DB cluster parameter group
Creates a snapshot of a DB cluster
Creates a new DB instance
Creates a new DB instance that acts as a read replica for an existing source DB instance
Creates a new DB parameter group
Creates a new DB proxy
Creates a DBProxyEndpoint
Creates a new DB security group
Creates a new DB shard group for Aurora Limitless Database
Creates a snapshot of a DB instance
Creates a new DB subnet group
Creates an RDS event notification subscription
Creates an Aurora global database spread across multiple Amazon Web Services Regions
Creates a zero-ETL integration with Amazon Redshift
create_option_group
create_tenant_database
delete_blue_green_deployment
delete_custom_db_engine_version
delete_db_cluster
delete_db_cluster_automated_backup
delete_db_cluster_endpoint
delete_db_cluster_parameter_group
delete_db_cluster_snapshot
delete_db_instance
delete_db_instance_automated_backup
delete_db_parameter_group
delete_db_proxy
delete_db_proxy_endpoint
delete_db_security_group
delete_db_shard_group
delete_db_snapshot
delete_db_subnet_group
delete_event_subscription
delete_global_cluster
delete_integration
delete_option_group
delete_tenant_database
deregister_db_proxy_targets
describe_account_attributes
describe_blue_green_deployments
describe_certificates
describe_db_cluster_automated_backups
describe_db_cluster_backtracks
describe_db_cluster_endpoints
describe_db_cluster_parameter_groups
describe_db_cluster_parameters
describe_db_clusters
describe_db_cluster_snapshot_attributes
describe_db_cluster_snapshots
describe_db_engine_versions
describe_db_instance_automated_backups
describe_db_instances
describe_db_log_files
describe_db_parameter_groups
describe_db_parameters
describe_db_proxies
describe_db_proxy_endpoints
describe_db_proxy_target_groups
describe_db_proxy_targets
describe_db_recommendations
describe_db_security_groups
describe_db_shard_groups

create_option_group
Creates a new option group
create_tenant_database
Creates a tenant database in a DB instance that uses the multi-tenant configuration
delete_blue_green_deployment
Deletes a blue/green deployment
delete_custom_db_engine_version
Deletes a custom engine version
delete_db_cluster
The DeleteDBCluster action deletes a previously provisioned DB cluster
delete_db_cluster_automated_backup
Deletes automated backups using the DbClusterResourceId value of the source DB cluster
delete_db_cluster_endpoint
Deletes a custom endpoint and removes it from an Amazon Aurora DB cluster
delete_db_cluster_parameter_group
Deletes a specified DB cluster parameter group
delete_db_cluster_snapshot
Deletes a DB cluster snapshot
delete_db_instance
Deletes a previously provisioned DB instance
delete_db_instance_automated_backup
Deletes automated backups using the DbInstanceResourceId value of the source DB instance
delete_db_parameter_group
Deletes a specified DB parameter group
delete_db_proxy
Deletes an existing DB proxy
delete_db_proxy_endpoint
Deletes a DBProxyEndpoint
delete_db_security_group
Deletes a DB security group
delete_db_shard_group
Deletes an Aurora Limitless Database DB shard group
delete_db_snapshot
Deletes a DB snapshot
delete_db_subnet_group
Deletes an RDS event notification subscription
delete_global_cluster
Deletes a global database cluster
delete_integration
Deletes a zero-ETL integration with Amazon Redshift
delete_option_group
Deletes an existing option group
delete_tenant_database
Deletes a tenant database from your DB instance
deregister_db_proxy_targets
Remove the association between one or more DBProxyTarget data structures and a DBProxyTargetGroup
describe_account_attributes
Lists all of the attributes for a customer account
describe_blue_green_deployments
Lists the set of certificate authority (CA) certificates provided by Amazon RDS
describe_certificates
Displays backups for both current and deleted DB clusters
describe_db_cluster_automated_backups
Returns information about backtracks for a DB cluster
describe_db_cluster_backtracks
Returns information about endpoints for an Amazon Aurora DB cluster
describe_db_cluster_endpoints
Returns a list of DBClusterParameterGroup descriptions
describe_db_cluster_parameter_groups
Returns the detailed parameter list for a particular DB cluster parameter group
describe_db_cluster_parameters
Describes existing Amazon Aurora DB clusters and Multi-AZ DB clusters
describe_db_clusters
Returns a list of DB cluster snapshot attribute names and values for a manually created snapshot
describe_db_cluster_snapshot_attributes
Describes the properties of specific versions of DB engines
describe_db_engine_versions
Displays backups for both current and deleted instances
describe_db_instances
Describes provisioned RDS instances
describe_db_instance_automated_backups
Returns a list of DB log files for the DB instance
describe_db_log_files
Returns a list of DBParameterGroup descriptions
describe_db_parameter_groups
Returns the detailed parameter list for a particular DB parameter group
describe_db_parameters
Returns information about DB proxies
describe_db_proxies
Returns information about DB proxy endpoints
describe_db_proxy_endpoints
Returns information about DB proxy target groups, represented by DBProxyTarget objects
describe_db_proxy_target_groups
Describes the recommendations to resolve the issues for your DB instance
describe_db_recommendations
Returns a list of DBSecurityGroup descriptions
describe_db_security_groups
Describes existing Aurora Limitless Database DB shard groups
describe_db_shard_groups
describe_db_snapshot_attributes
Returns a list of DB snapshot attribute names and values for a manual DB snapshot
describe_db_snapshots
Returns information about DB snapshots
describe_db_snapshot_tenant_databases
Describes the tenant databases that exist in a DB snapshot
describe_db_subnet_groups
Returns a list of DBSubnetGroup descriptions
describe_engine_default_cluster_parameters
Returns the default engine and system parameter information for the cluster
describe_engine_default_parameters
Returns the default engine and system parameter information for the specified database engine
describe_event_categories
Displays a list of categories for all event source types, or, if specified, for a specific event source type
describe_events
Returns events related to DB instances, DB clusters, DB parameter groups, DB snapshots, DB cluster snapshots, and RDS Proxies for the past 14 days
describe_event_subscriptions
Lists all the subscription descriptions for a customer account
describe_export_tasks
Returns information about a snapshot or cluster export to Amazon S3
describe_global_clusters
Returns information about Aurora global database clusters
describe_integrations
Describe one or more zero-ETL integrations with Amazon Redshift
describe_option_group_options
Describes all available options for the specified engine
describe_option_groups
Describes the available option groups
describe_orderable_db_instance_options
Describes the orderable DB instance options for a specified DB engine
describe_pending_maintenance_actions
Returns a list of resources (for example, DB instances) that have at least one pending maintenance action
describe_reserved_db_instances
Returns information about reserved DB instances for this account, or about a specified reserved DB instance
describe_reserved_db_instances_offerings
Lists available reserved DB instance offerings
describe_source_regions
Returns a list of the source Amazon Web Services Regions where the current Amazon Web Services Region can create a read replica, copy a DB snapshot from, or replicate automated backups from
describe_tenant_databases
Describes the tenant databases in a DB instance that uses the multi-tenant configuration
describe_valid_db_instance_modifications
You can call DescribeValidDBInstanceModifications to learn what modifications you can make to your DB instance
disable_http_endpoint
Disables the HTTP endpoint for the specified DB cluster
download_db_log_file_portion
Downloads all or a portion of the specified log file, up to 1 MB in size
disable_http_endpoint
Enables the HTTP endpoint for the DB cluster
disable_http_endpoint
Sets the capacity of an Aurora Serverless v1 DB cluster to a specific value
disable_http_endpoint
Modify the status of a custom engine version (CEV)
disable_http_endpoint
Modifies the settings of an Amazon Aurora DB cluster or a Multi-AZ DB cluster
disable_http_endpoint
Modifies the settings of a zero-ETL integration with Amazon Redshift
disable_http_endpoint
Modifies the settings of a DB instance
disable_http_endpoint
Modifies the parameters of a DB parameter group
disable_http_endpoint
Changes the audit policy state of a database activity stream to either locked or unlocked
disable_http_endpoint
Override the system-default Secure Sockets Layer/Transport Layer Security (SSL/TLS) certificate for Amazon RDS for new DB instances, or remove the override
disable_http_endpoint
Modify the properties of an endpoint in an Amazon Aurora DB cluster
disable_http_endpoint
Modifies the properties of a DBProxyTargetGroup
disable_http_endpoint
Modifies the recommendation status and recommended action status for the specified recommendation
disable_http_endpoint
Modifies the parameters of a DB cluster parameter group
disable_http_endpoint
Adds an attribute and values to, or removes an attribute and values from, a DB snapshot
disable_http_endpoint
Modifies settings for a DB instance
disable_http_endpoint
Modifies the properties of a DB instance
disable_http_endpoint
Changes the settings for an existing DB proxy
disable_http_endpoint
Changes the settings for an existing DB proxy endpoint
disable_http_endpoint
Modifies the properties of a RDS ProxyTargetGroup
disable_http_endpoint
Updates the recommendation status and recommended action status for the specified recommendation
disable_http_endpoint
Modifies the settings of an Aurora Limitless Database DB shard group
disable_http_endpoint
Updates a manual DB snapshot with a new engine version
disable_http_endpoint
Adds an attribute and values to, or removes an attribute and values from, a DB snapshot
disable_http_endpoint
Modifies an existing DB subnet group
disable_http_endpoint
Modifies an existing RDS event notification subscription
disable_http_endpoint
Modifies a setting for an Amazon Aurora global database cluster
disable_http_endpoint
Modifies a zero-ETL integration with Amazon Redshift
modify_option_group
modify_tenant_database
promote_read_replica
promote_read_replica_db_cluster
purchase_reserved_db_instances_offering
reboot_db_cluster
reboot_db_instance
reboot_db_shard_group
register_db_proxy_targets
remove_from_global_cluster
remove_role_from_db_cluster
remove_role_from_db_instance
remove_source_identifier_from_subscription
remove_tags_from_resource
reset_db_cluster_parameter_group
reset_db_parameter_group
restore_db_cluster_from_s3
restore_db_cluster_from_snapshot
restore_db_cluster_to_point_in_time
restore_db_instance_from_db_snapshot
restore_db_instance_from_s3
restore_db_instance_to_point_in_time
revoke_db_security_group_ingress
start_activity_stream
start_db_cluster
start_db_instance
start_db_instance_automated_backups_replication
start_export_task
stop_activity_stream
stop_db_cluster
stop_db_instance
stop_db_instance_automated_backups_replication
switchover_blue_green_deployment
switchover_global_cluster
switchover_read_replica

Modifies an existing option group
Modifies an existing tenant database in a DB instance
Promotes a read replica DB instance to a standalone DB instance
Promotes a read replica DB cluster to a standalone DB cluster
Purchases a reserved DB instance offering
You might need to reboot your DB cluster, usually for maintenance reasons
You might need to reboot your DB instance, usually for maintenance reasons
You might need to reboot your DB shard group, usually for maintenance reasons
Associate one or more DBProxyTarget data structures with a DBProxyTarget
Detaches an Aurora secondary cluster from an Aurora global database cluster
Removes the association of an Amazon Web Services Identity and Access Management (IAM) role from a DB cluster
Removes a source identifier from an existing RDS event notification subscription
Removes metadata tags from an RDS resource
Modifies the parameters of a DB cluster parameter group to the default value
Modifies the parameters of a DB parameter group to the engine/system default value
Creates an Amazon Aurora DB cluster from MySQL data stored in an Amazon S3 bucket
Creates a new DB cluster from a DB snapshot or DB cluster snapshot
Restores a DB cluster to an arbitrary point in time
Creates a new DB instance from a DB snapshot
Amazon Relational Database Service (Amazon RDS) supports importing a DB instance to an arbitrary point in time
Restores a DB instance to an arbitrary point in time
Revoke ingress from a DBSecurityGroup for previously authorized IP ranges or VPC security groups
Starts a database activity stream to monitor activity on the database
Starts an Amazon Aurora DB cluster that was stopped using the Amazon RDS console
Starts an Amazon RDS DB instance that was stopped using the Amazon RDS console
Enables replication of automated backups to a different Amazon Web Services Region
Starts an export of DB snapshot or DB cluster data to Amazon S3
Stops a database activity stream that was started using the Amazon RDS console
Stops an Amazon Aurora DB cluster
Stops an Amazon RDS DB instance
Stops automated backup replication for a DB instance
Switches over blue/green deployment
Switches the specified secondary DB cluster to be the new primary instance
Switches over an Oracle standby database in an Oracle Data Guard environment

Examples

```r
## Not run:
svc <- rds()
svc$add_role_to_db_cluster(
  Foo = 123
)

## End(Not run)
```
**AWS RDS DataService**

**Description**

RDS Data API

Amazon RDS provides an HTTP endpoint to run SQL statements on an Amazon Aurora DB cluster. To run these statements, you use the RDS Data API (Data API).

Data API is available with the following types of Aurora databases:

- Aurora PostgreSQL - Serverless v2, Serverless v1, and provisioned
- Aurora MySQL - Serverless v1 only

For more information about the Data API, see Using RDS Data API in the Amazon Aurora User Guide.

**Usage**

```r
rdsdataservice(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

**Arguments**

- `config`: Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
    - `creds`:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
    - `anonymous`: Set anonymous credentials.
  - `endpoint`: The complete URL to use for the constructed client.
  - `region`: The AWS Region used in instantiating the client.
  - `close_connection`: Immediately close all HTTP connections.
  - `timeout`: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - `s3_force_path_style`: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`. 
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials**
Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.

- **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

---

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- rdsdataservice(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  )
)
```
anonymous = "logical",
),
```
```
```

### Operations

- **batch_execute_statement**
  - Runs a batch SQL statement over an array of data
- **begin_transaction**
  - Starts a SQL transaction
- **commit_transaction**
  - Ends a SQL transaction started with the BeginTransaction operation and commits the changes
- **execute_sql**
  - Runs one or more SQL statements
- **execute_statement**
  - Runs a SQL statement against a database
- **rollback_transaction**
  - Performs a rollback of a transaction

### Examples

```r
## Not run:
svc <- rdsdataservice()
svc$batch_execute_statement(
  Foo = 123
)

## End(Not run)
```

---

**Description**

This is the *Recycle Bin API Reference*. This documentation provides descriptions and syntax for each of the actions and data types in Recycle Bin.

Recycle Bin is a resource recovery feature that enables you to restore accidentally deleted snapshots and EBS-backed AMIs. When using Recycle Bin, if your resources are deleted, they are retained in the Recycle Bin for a time period that you specify.

You can restore a resource from the Recycle Bin at any time before its retention period expires. After you restore a resource from the Recycle Bin, the resource is removed from the Recycle Bin, and you can then use it in the same way you use any other resource of that type in your account. If the retention period expires and the resource is not restored, the resource is permanently deleted from the Recycle Bin and is no longer available for recovery. For more information about Recycle Bin, see *Recycle Bin* in the *Amazon Elastic Compute Cloud User Guide*. 
Usage

recyclebin(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

config Optional configuration of credentials, endpoint, and/or region.

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
    - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.
- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

credentials Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

svc <- recyclebin(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)

Operations

create_rule Creates a Recycle Bin retention rule
delete_rule Deletes a Recycle Bin retention rule
get_rule Gets information about a Recycle Bin retention rule
list_rules Lists the Recycle Bin retention rules in the Region
list_tags_for_resource Lists the tags assigned to a retention rule
lock_rule Locks a retention rule
tag_resource Assigns tags to the specified retention rule
unlock_rule Unlocks a retention rule
untag_resource Unassigns a tag from a retention rule
update_rule Updates an existing Recycle Bin retention rule
Examples

```r
## Not run:
svc <- recyclebin()
svc$create_rule(
   Foo = 123
)
## End(Not run)
```

Description

Overview

This is an interface reference for Amazon Redshift. It contains documentation for one of the programming or command line interfaces you can use to manage Amazon Redshift clusters. Note that Amazon Redshift is asynchronous, which means that some interfaces may require techniques, such as polling or asynchronous callback handlers, to determine when a command has been applied. In this reference, the parameter descriptions indicate whether a change is applied immediately, on the next instance reboot, or during the next maintenance window. For a summary of the Amazon Redshift cluster management interfaces, go to Using the Amazon Redshift Management Interfaces.

Amazon Redshift manages all the work of setting up, operating, and scaling a data warehouse: provisioning capacity, monitoring and backing up the cluster, and applying patches and upgrades to the Amazon Redshift engine. You can focus on using your data to acquire new insights for your business and customers.

If you are a first-time user of Amazon Redshift, we recommend that you begin by reading the Amazon Redshift Getting Started Guide.

If you are a database developer, the Amazon Redshift Database Developer Guide explains how to design, build, query, and maintain the databases that make up your data warehouse.

Usage

```r
redshift(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token

---

**redshift**

*Amazon Redshift*
– **profile**: The name of a profile to use. If not given, then the default profile is used.
– **anonymous**: Set anonymous credentials.

**endpoint**: The complete URL to use for the constructed client.

**region**: The AWS Region used in instantiating the client.

**close_connection**: Immediately close all HTTP connections.

**timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

**s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.

**sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials** Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

## Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

## Service syntax

```r
svc <- redshift(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    ...,
  ),
  ...,
)
```
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
  creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
  ),
  profile = "string",
  anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

accept_reserved_node_exchange
add_partner
associate_data_share_consumer
authorize_cluster_security_group_ingress
authorize_data_share
authorize_endpoint_access
authorize_snapshot_access
batch_delete_cluster_snapshots
batch_modify_cluster_snapshots
cancel_resize
copy_cluster_snapshot
create_authentication_profile
create_cluster
create_cluster_parameter_group
create_cluster_security_group
create_cluster_snapshot
create_cluster_subnet_group
create_custom_domain_association
create_endpoint_access
create_event_subscription
create_hsm_client_certificate
create_hsm_configuration
create_redshift_idc_application
create_scheduled_action
create_snapshot_copy_grant
create_snapshot_schedule
create_tags
create_usage_limit
deauthorize_data_share

Exchanges a DC1 Reserved Node for a DC2 Reserved Node with no changes to the configuration (term, payment type, or number of nodes) and no additional costs.

Adds a partner integration to a cluster.

From a datashare consumer account, associates a datashare with the account (AssociateEntireAccount) or the specified namespace (ConsumerArn).

Adds an inbound (ingress) rule to an Amazon Redshift security group.

From a data producer account, authorizes the sharing of a datashare with one or more consumer accounts or managing entities.

Grants access to a cluster.

Authorizes the specified Amazon Web Services account to restore the specified snapshot.

Deletes a set of cluster snapshots.

Modifies the settings for a set of cluster snapshots.

Cancels a resize operation for a cluster.

Copies the specified automated cluster snapshot to a new manual cluster snapshot.

Creates an authentication profile with the specified parameters.

Creates a new cluster with the specified parameters.

Creates an Amazon Redshift parameter group.

Creates a new Amazon Redshift security group.

Creates a manual snapshot of the specified cluster.

Creates a new Amazon Redshift subnet group.

Used to create a custom domain name for a cluster.

Creates a Redshift-managed VPC endpoint.

Creates an Amazon Redshift event notification subscription.

Creates an HSM client certificate that an Amazon Redshift cluster will use.

Creates an HSM configuration that contains the information required by the cluster.

Creates an Amazon Redshift application for use with IAM Identity Center.

Creates a scheduled action.

Creates a snapshot copy grant that permits Amazon Redshift to use an KMS key to encrypt snapshots.

Create a snapshot schedule that can be associated to a cluster and which defines the time and frequency at which the snapshots must be taken.

Adds tags to a cluster.

Creates a usage limit for a specified Amazon Redshift feature on a cluster.

From a datashare producer account, removes authorization from the specified datashare.
delete_authentication_profile
delete_cluster
delete_cluster_parameter_group
delete_cluster_security_group
delete_cluster_snapshot
delete_cluster_subnet_group
delete_custom_domain_association
delete_endpoint_access
delete_event_subscription
delete_hsm_client_certificate
delete_hsm_configuration
delete_partner
delete_redshift_idc_application
delete_resource_policy
delete_scheduled_action
delete_snapshot_copy_grant
delete_snapshot_schedule
delete_tags
delete_usage_limit
describe_account_attributes
describe_authentication_profiles
describe_cluster_db_revisions
describe_cluster_parameter_groups
describe_cluster_parameters
describe_clusters
describe_cluster_security_groups
describe_cluster_snapshots
describe_cluster_subnet_groups
describe_cluster_tracks
describe_custom_domain_associations
describe_data_shares
describe_data_shares_for_consumer
describe_data_shares_for_producer
describe_default_cluster_parameters
describe_endpoint_access
describe_endpoint_authorization
describe_event_categories
describe_events
describe_event_subscriptions
describe_hsm_client_certificates
describe_hsm_configurations
describe_inbound_integrations
describe_logging_status
describe_node_configuration_options
describe_orderable_cluster_options
describe_partners
describe_redshift_idc_applications

Deletes an authentication profile
Deletes a previously provisioned cluster without its final snapshot being created
Deletes a specified Amazon Redshift parameter group
Deletes an Amazon Redshift security group
Deletes the specified manual snapshot
Deletes the specified cluster subnet group
Contains information about deleting a custom domain association for a cluster
Deletes a Redshift-managed VPC endpoint
Deletes an Amazon Redshift event notification subscription
Deletes the specified HSM client certificate
Deletes the specified Amazon Redshift HSM configuration
Deletes a partner integration from a cluster
Deletes an Amazon Redshift IAM Identity Center application
Deletes the resource policy for a specified resource
Deletes a scheduled action
Deletes the specified snapshot copy grant
Deletes a snapshot schedule
Deletes tags from a resource
Deletes a usage limit from a cluster
Returns a list of attributes attached to an account
Describes an authentication profile
Returns an array of ClusterDbRevision objects
Returns a list of Amazon Redshift parameter groups, including parameter groups you created and the default parameter group
Returns a detailed list of parameters contained within the specified Amazon Redshift parameter group
Returns properties of provisioned clusters including general cluster properties
Returns information about Amazon Redshift security groups
Returns one or more snapshot objects, which contain metadata about your snapshots
Returns one or more cluster subnet group objects, which contain metadata about your cluster subnet groups
Returns a list of all the available maintenance tracks
Returns descriptions of the available Amazon Redshift cluster versions
Contains information about custom domain associations for a cluster
Shows the status of any inbound or outbound datashares available in the account
Returns a list of datashares where the account identifier being called is a consumer
Returns a list of datashares when the account identifier being called is a producer
Returns a list of parameter settings for the specified parameter group family
Describes a Redshift-managed VPC endpoint
Describes an endpoint authorization
Displays a list of event categories for all event source types, or for a specific event source type
Returns events related to clusters, security groups, snapshots, and parameter groups
Lists descriptions of all the Amazon Redshift event notification subscriptions
Returns information about the specified HSM client certificate
Returns information about the specified Amazon Redshift HSM configuration
Returns a list of inbound integrations
Describes whether information, such as queries and connection attempts, is logged for the specified Amazon Redshift cluster
Returns properties of possible node configurations such as node type, replication method, and instance type
Returns a list of orderable cluster options
Returns information about the partner integrations defined for a cluster
Lists the Amazon Redshift IAM Identity Center applications
describe_reserved_node_exchange_status
describe_reserved_node_offerings
describe_reserved_nodes
describe_scheduled_actions
describe_snapshot_copy_grants
describe_snapshot_schedules
describe_storage
describe_table_restore_status
describe_tags
describe_usage_limits
disable_logging
disable_snapshot_copy
disassociate_data_share_consumer
enable_logging
enable_snapshot_copy
failover_primary_compute
get_cluster_credentials
get_cluster_credentials_with_iam
get_reserved_node_exchange_configuration_options
get_reserved_node_exchange_offerings
get_resource_policy
list_recommendations
modify_aqua_configuration
modify_authentication_profile
modify_cluster
modify_cluster_db_revision
modify_cluster_iam_roles
modify_cluster_maintenance
modify_cluster_parameter_group
modify_cluster_snapshot
modify_cluster_snapshot_schedule
modify_cluster_subnet_group
modify_custom_domain_association
modify_endpoint_access
modify_event_subscription
modify_redshift_idc_application
modify_scheduled_action
modify_snapshot_copy_retention_period
modify_snapshot_schedule
modify_usage_limit
pause_cluster
purchase_reserved_node_offering
put_resource_policy
reboot_cluster
reject_data_share
reset_cluster_parameter_group
resize_cluster

Returns exchange status details and associated metadata for a reserved-node exchange.
Returns a list of the available reserved node offerings by Amazon Redshift.
Returns the descriptions of the reserved nodes.
Returns the last resize operation for the specified cluster.
Describes properties of scheduled actions.
Returns a list of snapshot copy grants owned by the Amazon Web Services account.
Returns a list of snapshot schedules.
Returns account level backups storage size and provisional storage.
Lists the status of one or more table restore requests made using the Redshift REST API.
Returns a list of tags.
Shows usage limits on a cluster.

View additional actions for Amazon Redshift.
You can use the Amazon Redshift Data API to run queries on Amazon Redshift tables. You can run SQL statements, which are committed if the statement succeeds.

For more information about the Amazon Redshift Data API and CLI usage examples, see Using the Amazon Redshift Data API in the Amazon Redshift Management Guide.

### Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
* access_key_id: AWS access key ID
* secret_access_key: AWS secret access key
* session_token: AWS temporary session token

- profile: The name of a profile to use. If not given, then the default profile is used.

- anonymous: Set anonymous credentials.

- endpoint: The complete URL to use for the constructed client.
- region: The AWS Region used in instantiating the client.
- close_connection: Immediately close all HTTP connections.
- timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

- s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

- sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter

- creds:
  - access_key_id: AWS access key ID
  - secret_access_key: AWS secret access key
  - session_token: AWS temporary session token

- profile: The name of a profile to use. If not given, then the default profile is used.

- anonymous: Set anonymous credentials.

documentation Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- redshiftdataapiservice(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    )
  )
)```
redshiftdataapiservice

),
endpoint = "string",
region = "string",
close_connection = "logical",
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
  creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
  ),
  profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

batch_execute_statement  Runs one or more SQL statements, which can be data manipulation language (DML) or data definition language (DDL)
cancel_statement        Cancels a running query
describe_statement      Describes the details about a specific instance when a query was run by the Amazon Redshift Data API
describe_table          Describes the detailed information about a table from metadata in the cluster
execute_statement       Runs an SQL statement, which can be data manipulation language (DML) or data definition language (DDL)get_statement_result    Fetches the temporarily cached result of an SQL statement
list_databases          List the databases in a cluster
list_schemas            Lists the schemas in a database
list_statements         List of SQL statements
list_tables             List the tables in a database

Examples

## Not run:
svc <- redshiftdataapiservice()
svc$batch_execute_statement(
  Foo = 123
)

## End(Not run)
Description

This is an interface reference for Amazon Redshift Serverless. It contains documentation for one of the programming or command line interfaces you can use to manage Amazon Redshift Serverless. Amazon Redshift Serverless automatically provisions data warehouse capacity and intelligently scales the underlying resources based on workload demands. Amazon Redshift Serverless adjusts capacity in seconds to deliver consistently high performance and simplified operations for even the most demanding and volatile workloads. Amazon Redshift Serverless lets you focus on using your data to acquire new insights for your business and customers.

To learn more about Amazon Redshift Serverless, see What is Amazon Redshift Serverless.

Usage

```python
redshiftserverless(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

Arguments

- `config`: Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
    - `creds`:
      * `access_key_id`: AWS access key ID
      * `secret_access_key`: AWS secret access key
      * `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
    - `anonymous`: Set anonymous credentials.
  - `endpoint`: The complete URL to use for the constructed client.
  - `region`: The AWS Region used in instantiating the client.
  - `close_connection`: Immediately close all HTTP connections.
  - `timeout`: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - `s3_force_path_style`: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
  - `sts_regional_endpoint`: Set sts regional endpoint resolver to regional or legacy `https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-extensions.html`
credentials  Optional credentials shorthand for the config parameter
  
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  
  • profile: The name of a profile to use. If not given, then the default profile is used.

  • anonymous: Set anonymous credentials.

document

elements

traits

tasks

credentials  Optional credentials shorthand for the config parameter
  
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  
  • profile: The name of a profile to use. If not given, then the default profile is used.

  • anonymous: Set anonymous credentials.

document

elements

traits

tasks

endpoint  Optional shorthand for complete URL to use for the constructed client.

region  Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

svc <- redshiftserverless(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
region = "string"
}

Operations

convert_recovery_point_to_snapshot
create_custom_domain_association
create_endpoint_access
create_namespace
create_scheduled_action
create_snapshot
create_snapshot_copy_configuration
create_usage_limit
create_workgroup
delete_custom_domain_association
delete_endpoint_access
delete_namespace
delete_resource_policy
delete_scheduled_action
delete_snapshot
delete_snapshot_copy_configuration
delete_usage_limit
delete_workgroup
get_credentials
get_custom_domain_association
get_endpoint_access
get_namespace
get_recovery_point
get_resource_policy
get_scheduled_action
get_snapshot
get_table_restore_status
get_usage_limit
get_workgroup
list_custom_domain_associations
list_endpoint_access
list_namespaces
list_recovery_points
list_scheduled_actions
list_snapshot_copy_configurations
list_snapshots
list_table_restore_status
list_tags_for_resource
list_usage_limits
list_workgroups
put_resource_policy
restore_from_recovery_point
restore_from_snapshot

Converts a recovery point to a snapshot
Converts a custom domain association for Amazon Redshift Serverless
Creates an Amazon Redshift Serverless managed VPC endpoint
Creates a namespace in Amazon Redshift Serverless
Creates a scheduled action
Creates a snapshot of all databases in a namespace
Creates a snapshot copy configuration that lets you copy snapshots to another Amazon Web Services Region
Creates a usage limit for a specified Amazon Redshift Serverless usage type
Creates an workgroup in Amazon Redshift Serverless
Deletes a custom domain association for Amazon Redshift Serverless
Deletes an Amazon Redshift Serverless managed VPC endpoint
Deletes a namespace from Amazon Redshift Serverless
Deletes the specified resource policy
Deletes a scheduled action
Deletes a snapshot from Amazon Redshift Serverless
Deletes a snapshot copy configuration
Deletes a usage limit from Amazon Redshift Serverless
Deletes a workgroup
Returns a database user name and temporary password with temporary authorization to log in to Amazon Redshift Serverless
Gets information about a specific custom domain association
Returns information, such as the name, about a VPC endpoint
Returns information about a namespace in Amazon Redshift Serverless
Returns information about a recovery point
Returns a resource policy
Returns information about a scheduled action
Returns information about a specific snapshot
Returns information about a TableRestoreStatus object
Returns information about a usage limit
Returns information about a specific workgroup
Returns an array of EndpointAccess objects and relevant information
Returns information about a list of specified namespaces
Returns an array of recovery points
Returns a list of scheduled actions
Returns a list of snapshot copy configurations
Returns a list of snapshots
Returns information about an array of TableRestoreStatus objects
Lists the tags assigned to a resource
Lists all usage limits within Amazon Redshift Serverless
Returns information about a list of specified workgroups
Creates or updates a resource policy
Restore the data from a recovery point
Restores a namespace from a snapshot
### Examples

```r
## Not run:
svc <- redshiftserverless()
svc$convert_recovery_point_to_snapshot(
  Foo = 123
)
## End(Not run)
```

---

### Description

This is the API Reference for Amazon Rekognition Image, Amazon Rekognition Custom Labels, Amazon Rekognition Stored Video, Amazon Rekognition Streaming Video. It provides descriptions of actions, data types, common parameters, and common errors.

**Amazon Rekognition Image**

- associate_faces
- compare_faces
- create_collection
- create_user
- delete_collection
- delete_faces
- delete_user
- describe_collection
- detect_faces
- detect_labels
- detect_moderation_labels
- detect_protective_equipment
- detect_text
- disassociate_faces
- get_celebrity_info
- get_media_analysis_job
- index_faces
- list_collections
- ListMediaAnalysisJob
- list_faces
- list_users
- recognize_celebrities
- search_faces
- search_faces_by_image
- search_users
- search_users_by_image
- start_media_analysis_job

**Amazon Rekognition Custom Labels**

- copy_project_version
- create_dataset
- create_project
- create_project_version
- delete_dataset
- delete_project
- delete_project_policy
- delete_project_version
- describe_dataset
- describe_projects
- describe_project_versions
- detect_custom_labels
- distribute_dataset_entries
- list_dataset_entries
- list_dataset_labels
- list_project_policies
- put_project_policy
- start_project_version
• stop_project_version
• update_dataset_entries

Amazon Rekognition Video Stored Video
• get_celebrity_recognition
• get_content_moderation
• get_face_detection
• get_face_search
• get_label_detection
• get_person_tracking
• get_segment_detection
• get_text_detection
• start_celebrity_recognition
• start_content_moderation
• start_face_detection
• start_face_search
• start_label_detection
• start_person_tracking
• start_segment_detection
• start_text_detection

Amazon Rekognition Video Streaming Video
• create_stream_processor
• delete_stream_processor
• describe_stream_processor
• list_stream_processors
• start_stream_processor
• stop_stream_processor
• update_stream_processor

Usage
rekognition(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
Arguments

config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- rekognition(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
```

secret_access_key = "string",
    session_token = "string"
),
  profile = "string",
  anonymous = "logical"
),
  endpoint = "string",
  region = "string",
  close_connection = "logical",
  timeout = "numeric",
  s3_force_path_style = "logical",
  sts_regional_endpoint = "string"
),
credentials = list(
  creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
  ),
  profile = "string",
  anonymous = "logical"
),
  endpoint = "string",
  region = "string"
)

Operations

associate_faces Associates one or more faces with an existing UserID
compare_faces Compares a face in the source input image with each of the 100 largest faces detected in the target input image
copy_project_version This operation applies only to Amazon Rekognition Custom Labels
create_collection Creates a collection in an AWS Region
create_dataset This operation applies only to Amazon Rekognition Custom Labels
create_face_liveness_session This API operation initiates a Face Liveness session
create_project Creates a new Amazon Rekognition project
create_project_version Creates a new version of Amazon Rekognition project (like a Custom Labels model or a custom adapter)
create_stream_processor Creates an Amazon Rekognition stream processor that you can use to detect and recognize faces
create_user Creates a new User within a collection specified by CollectionId
delete_collection Deletes the specified collection
delete_dataset This operation applies only to Amazon Rekognition Custom Labels
delete_faces Deletes faces from a collection
delete_project Deletes a Amazon Rekognition project
delete_project_policy This operation applies only to Amazon Rekognition Custom Labels
delete_project_version Deletes a Rekognition project model or project version, like a Amazon Rekognition Custom Labels model or a custom adapter
delete_stream_processor Deletes the stream processor identified by Name
delete_user Deletes the specified UserID within the collection
describe_collection Describes the specified collection
describe_dataset This operation applies only to Amazon Rekognition Custom Labels
describe_projects
describe_project_versions
describe_stream_processor
detect_custom_labels
detect_faces
detect_labels
detect_moderation_labels
detect_protective_equipment
detect_text
disassociate_faces
distribute_dataset_entries
get_celebrity_info
get_celebrity_recognition
get_content_moderation
get_face_detection
get_face_liveness_session_results
get_face_search
get_label_detection
get_media_analysis_job
get_person_tracking
get_segment_detection
get_text_detection
index_faces
list_collections
list_dataset_entries
list_dataset_labels
list_faces
list_media_analysis_jobs
list_project_policies
list_stream_processors
list_tags_for_resource
list_users
put_project_policy
recognize_celebrities
search_faces
search_faces_by_image
search_users
search_users_by_image
start_celebrity_recognition
start_content_moderation
start_face_detection
start_face_search
start_label_detection
start_media_analysis_job
start_person_tracking
start_project_version
start_segment_detection
start_stream_processor

- **describe_projects**
  - Gets information about your Rekognition projects
  - Lists and describes the versions of an Amazon Rekognition project
- **describe_project_versions**
  - Provides information about a stream processor created by CreateStreamProcessor
  - This operation applies only to Amazon Rekognition Custom Labels
- **describe_stream_processor**
  - Detects faces within an image that is provided as input
  - Detects unsafe content in a specified JPEG or PNG format image
  - Detects Personal Protective Equipment (PPE) worn by people detected in an image
  - Detects text in the input image and converts it into machine-readable text
- **detect_custom_labels**
  - Removes the association between a Face supplied in an array of FaceIds and the User
  - This operation applies only to Amazon Rekognition Custom Labels
- **detect_faces**
  - Gets the name and additional information about a celebrity based on their Amazon Rekognition ID
  - Gets the celebrity recognition results for a Amazon Rekognition Video analysis started by
  - Gets inappropriate, unwanted, or offensive content analysis results for a Amazon Rekognition Video analysis started by
  - Gets face detection results for a Amazon Rekognition Video analysis started by
  - Retrieves the results of a specific Face Liveness session
  - Gets the face search results for Amazon Rekognition Video face search started by
  - Gets the label detection results of a Amazon Rekognition Video analysis started by
  - Retrieves the results for a given media analysis job
- **detect_labels**
  - Gets the path tracking results of a Amazon Rekognition Video analysis started by
  - Gets the segment detection results of a Amazon Rekognition Video analysis started by
  - Gets the text detection results of a Amazon Rekognition Video analysis started by
  - Detects faces in the input image and adds them to the specified collection
  - Returns list of collection IDs in your account
  - This operation applies only to Amazon Rekognition Custom Labels
  - This operation applies only to Amazon Rekognition Custom Labels
  - Returns metadata for faces in the specified collection
  - This operation applies only to Amazon Rekognition Custom Labels
  - Returns a list of media analysis jobs
  - This operation applies only to Amazon Rekognition Custom Labels
  - Gets a list of stream processors that you have created with CreateStreamProcessor
  - Returns a list of tags in an Amazon Rekognition collection, stream processor, or Custom Labels model
  - Returns metadata of the User such as UserID in the specified collection
  - This operation applies only to Amazon Rekognition Custom Labels
  - Returns an array of celebrities recognized in the input image
  - For a given input face ID, searches for matching faces in the collection the face belongs to
  - For a given input image, first detects the largest face in the image, and then searches the specified collection
  - Searches for UserIDs within a collection based on a FaceId or UserId
  - Searches for User IDs using a supplied image
  - Starts asynchronous recognition of celebrities in a stored video
  - Starts asynchronous detection of inappropriate, unwanted, or offensive content in a stored video
  - Starts asynchronous detection of faces in a stored video
  - Starts the asynchronous search for faces in a collection that match the faces of persons detected in a stored video
  - Starts asynchronous detection of labels in a stored video
  - Initiates a new media analysis job
  - Starts the asynchronous tracking of a person’s path in a stored video
  - This operation applies only to Amazon Rekognition Custom Labels
  - Starts asynchronous detection of segment detection in a stored video
  - Starts processing a stream processor
Description

Resilience Hub helps you proactively prepare and protect your Amazon Web Services applications from disruptions. It offers continual resiliency assessment and validation that integrates into your software development lifecycle. This enables you to uncover resiliency weaknesses, ensure recovery time objective (RTO) and recovery point objective (RPO) targets for your applications are met, and resolve issues before they are released into production.

Usage

```r
caller$associate_faces(
  ClientRequestToken = "550e8400-e29b-41d4-a716-446655440002",
  CollectionId = "MyCollection",
  FaceIds = list(  
    "f5817d37-94f6-4335-8f70-4d93d6c4a89b",
    "851cb847-dccc-4fe9-9309-9f4805967855",
    "35ebbb41-7f67-4263-908d-dd0ecba05ab9"
  ),
  UserId = "DemoUser",
  UserMatchThreshold = 70L
)
```
Arguments

Optional configuration of credentials, endpoint, and/or region.

- **credentials**: 
  - **creds**: 
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials**

Optional credentials shorthand for the config parameter

- **creds**: 
  - `access_key_id`: AWS access key ID
  - `secret_access_key`: AWS secret access key
  - `session_token`: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint**

Optional shorthand for complete URL to use for the constructed client.

**region**

Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- resiliencehub(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"),
    profile = "string")
  
```
secret_access_key = "string",
    session_token = "string",

    profile = "string",
    anonymous = "logical",

    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
),

credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),

    endpoint = "string",
    region = "string"
)

Operations

add_draft_app_version_resource_mappings
batch_update_recommendation_status
create_app
create_app_version_app_component
create_app_version_resource
create_recommendation_template
create_resiliency_policy
delete_app
delete_app_assessment
delete_app_input_source
delete_app_version_app_component
delete_app_version_resource
delete_recommendation_template
delete_resiliency_policy
describe_app
describe_app_assessment
describe_app_version
describe_app_version_app_component
describe_app_version_resource
describe_app_version_resources_resolution_status

Adds the source of resource-maps to the draft version of an application
Enables you to include or exclude one or more operational recommendations
Creates an Resilience Hub application
Creates a new Application Component in the Resilience Hub application
Adds a resource to the Resilience Hub application and assigns it to the specified Application Components
Creates a resiliency policy for an application
Deletes an Resilience Hub application
Deletes an Resilience Hub application assessment
Deletes the input source and all of its imported resources from the Resilience Hub application
Deletes an Application Component from the Resilience Hub application
Deletes a resource from the Resilience Hub application
Deletes a recommendation template
Deletes a resiliency policy
Describes an Resilience Hub application
Describes an assessment for an Resilience Hub application
Describes the Resilience Hub application version
Describes an Application Component in the Resilience Hub application
Describes a resource of the Resilience Hub application
Returns the resolution status for the specified resolution identifier for an application
describe_app_version_template
describe_draft_app_version_resources_import_status
describe_resiliency_policy
import_resources_to_draft_app_version
list_alarm_recommendations
list_app_assessment_compliance_drifts
list_app_assessments
list_app_component_compliances
list_app_component_recommendations
list_app_input_sources
list_apps
list_app_version_app_components
list_app_version_resource_mappings
list_app_version_resources
list_app_versions
list_recommendation_templates
list_resiliency_policies
list_sop_recommendations
list_suggested_resiliency_policies
list_tags_for_resource
list_test_recommendations
list_unsupported_app_version_resources
publish_app_version
put_draft_app_version_template
remove_draft_app_version_resource_mappings
resolve_app_version_resources
start_app_assessment
tag_resource
untag_resource
update_app
update_app_version
update_app_version_app_component
update_app_version_resource
update_resiliency_policy

Describes details about a Resilience Hub application
Describes the status of importing resources to an application version
Describes a specified resiliency policy for an Resilience Hub application
Imports resources to Resilience Hub application draft version from different sources
Lists the alarm recommendations for an Resilience Hub application
List of compliance drifts that were detected while running an assessment
Lists the assessments for an Resilience Hub application
Lists the compliances for an Resilience Hub Application Component
Lists the recommendations for an Resilience Hub Application Component
Lists all the input sources of the Resilience Hub application
Lists your Resilience Hub applications
Lists all the Application Components in the Resilience Hub application
Lists how the resources in an application version are mapped/sourced
Lists all the resources in an Resilience Hub application
Lists the different versions for the Resilience Hub applications
Lists the recommendation templates for the Resilience Hub application
Lists the resiliency policies for the Resilience Hub applications
Lists the standard operating procedure (SOP) recommendations for the
Lists the suggested resiliency policies for the Resilience Hub applications
Lists the tags for your resources in your Resilience Hub applications
Lists the test recommendations for the Resilience Hub application
Lists the resources that are not currently supported in Resilience Hub
Publishes a new version of a specific Resilience Hub application
Adds or updates the app template for an Resilience Hub application draft
Removes resource mappings from a draft application version
Resolves the resources for an application version
Creates a new application assessment for an application
Applies one or more tags to a resource
Removes one or more tags from a resource
Updates an application
Updates the Resilience Hub application version
Updates an existing Application Component in the Resilience Hub application
Updates the resource details in the Resilience Hub application
Updates a resiliency policy

Examples

```r
## Not run:
svc <- resiliencehub()
svc$add_draft_app_version_resource_mappings(
  Foo = 123
)

## End(Not run)
```
Description

Amazon Web Services Resource Explorer is a resource search and discovery service. By using Resource Explorer, you can explore your resources using an internet search engine-like experience. Examples of resources include Amazon Relational Database Service (Amazon RDS) instances, Amazon Simple Storage Service (Amazon S3) buckets, or Amazon DynamoDB tables. You can search for your resources using resource metadata like names, tags, and IDs. Resource Explorer can search across all of the Amazon Web Services Regions in your account in which you turn the service on, to simplify your cross-Region workloads.

Resource Explorer scans the resources in each of the Amazon Web Services Regions in your Amazon Web Services account in which you turn on Resource Explorer. Resource Explorer creates and maintains an index in each Region, with the details of that Region’s resources.

You can search across all of the indexed Regions in your account by designating one of your Amazon Web Services Regions to contain the aggregator index for the account. When you promote a local index in a Region to become the aggregator index for the account, Resource Explorer automatically replicates the index information from all local indexes in the other Regions to the aggregator index. Therefore, the Region with the aggregator index has a copy of all resource information for all Regions in the account where you turned on Resource Explorer. As a result, views in the aggregator index Region include resources from all of the indexed Regions in your account.

For more information about Amazon Web Services Resource Explorer, including how to enable and configure the service, see the Amazon Web Services Resource Explorer User Guide.

Usage

```r
resourceexplorer(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:  
    - **creds**:  
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
• **endpoint**: The complete URL to use for the constructed client.
• **region**: The AWS Region used in instantiating the client.
• **close_connection**: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials** Optional credentials shorthand for the config parameter

• **creds**:
  – **access_key_id**: AWS access key ID
  – **secret_access_key**: AWS secret access key
  – **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.
• **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
cvc <- resourceexplorer(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  )
)
```
resourceexplorer

),
credentials = list(
  creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
  ),
  profile = "string",
  anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

associate_default_view
batch_get_view
create_index
create_view
delete_index
delete_view
disassociate_default_view
get_account_level_service_configuration
get_default_view
get_index
get_view
list_indexes
list_indexes_for_members
list_supported_resource_types
list_tags_for_resource
list_views
search
tag_resource
untag_resource
update_index_type
update_view

Sets the specified view as the default for the Amazon Web Services Region in which you called this operation.

Retrieves details about a list of views.

Turns on Amazon Web Services Resource Explorer in the Amazon Web Services Region in which you called this operation.

Creates a view that users can query by using the Search operation.

Deletes the specified index and turns off Amazon Web Services Resource Explorer in the Amazon Web Services Region.

Deletes the specified view.

After you call this operation, the affected Amazon Web Services Region no longer has a default view.

Retrieves the status of your account’s Amazon Web Services service access, and validates the service linked role required to access the multi-account search feature.

Retrieves details about the view that is the default for the Amazon Web Services Region in which you called this operation.

Retrieves details about the Amazon Web Services Resource Explorer index in the Amazon Web Services Region in which you called this operation.

Retrieves details of the specified view.

Retrieves a list of all of the indexes in Amazon Web Services Regions that are currently collecting resource information for Amazon Web Services Resource Explorer.

Retrieves a list of a member’s indexes in all Amazon Web Services Regions that are currently collecting resource information for Amazon Web Services Resource Explorer.

Retrieves a list of all resource types currently supported by Amazon Web Services Resource Explorer.

Lists the tags that are attached to the specified resource.

Retrieves a list of all of the indexes in Amazon Web Services Regions that are currently collecting resource information for Amazon Web Services Resource Explorer.

Lists the Amazon resource names (ARNs) of the views available in the Amazon Web Services Region.

Searches for resources and displays details about all resources that match the specified criteria.

Adds one or more tag key and value pairs to an Amazon Web Services Resource Explorer index.

Removes one or more tag key and value pairs from an Amazon Web Services Resource Explorer index.

Changes the type of the index from one of the following types to the other.

Modifies some of the details of a view.

Examples

```r
## Not run:
svc <- resourceexplorer()
svc$associate_default_view(
  Foo = 123
)

## End(Not run)
```
Resource Groups lets you organize Amazon Web Services resources such as Amazon Elastic Compute Cloud instances, Amazon Relational Database Service databases, and Amazon Simple Storage Service buckets into groups using criteria that you define as tags. A resource group is a collection of resources that match the resource types specified in a query, and share one or more tags or portions of tags. You can create a group of resources based on their roles in your cloud infrastructure, lifecycle stages, regions, application layers, or virtually any criteria. Resource Groups enable you to automate management tasks, such as those in Amazon Web Services Systems Manager Automation documents, on tag-related resources in Amazon Web Services Systems Manager. Groups of tagged resources also let you quickly view a custom console in Amazon Web Services Systems Manager that shows Config compliance and other monitoring data about member resources.

To create a resource group, build a resource query, and specify tags that identify the criteria that members of the group have in common. Tags are key-value pairs.

For more information about Resource Groups, see the Resource Groups User Guide.

Resource Groups uses a REST-compliant API that you can use to perform the following types of operations.

- Create, Read, Update, and Delete (CRUD) operations on resource groups and resource query entities
- Applying, editing, and removing tags from resource groups
- Resolving resource group member ARNs so they can be returned as search results
- Getting data about resources that are members of a group
- Searching Amazon Web Services resources based on a resource query

Usage

```r
resourcegroups(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

Arguments

- `config`: Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
    - `creds`:
      - `access_key_id`: AWS access key ID
* secret_access_key: AWS secret access key
* session_token: AWS temporary session token
  – profile: The name of a profile to use. If not given, then the default profile is used.
  – anonymous: Set anonymous credentials.

• endpoint: The complete URL to use for the constructed client.
• region: The AWS Region used in instantiating the client.
• close_connection: Immediately close all HTTP connections.
• timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.
region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```
svc <- resourcegroups(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string"
  ),
  region = "string"
)
```
resourcegroups

region = "string",
close_connection = "logical",
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

create_group    Creates a resource group with the specified name and description
delete_group   Deletes the specified resource group
get_account_settings Retrieves the current status of optional features in Resource Groups
get_group       Returns information about a specified resource group
get_group_configuration Retrieves the service configuration associated with the specified resource group
get_group_query  Retrieves the resource query associated with the specified resource group
group_resources Adds the specified resources to the specified group
list_group_resources Returns a list of ARNs of the resources that are members of a specified resource group
list_groups     Returns a list of existing Resource Groups in your account
put_group_configuration Attaches a service configuration to the specified group
search_resources Returns a list of Amazon Web Services resource identifiers that matches the specified query
tag            Adds tags to a resource group with the specified ARN
ungroup_resources Removes the specified resources from the specified group
untag           Deletes tags from a specified resource group
update_account_settings Turns on or turns off optional features in Resource Groups
update_group    Updates the description for an existing group
update_group_query  Updates the resource query of a group

Examples

## Not run:
svc <- resourcegroups()
svc$create_group(
    Foo = 123
)
AWS Resource Groups Tagging API

Description

Resource Groups Tagging API

Usage

resourcegroupstaggingapi(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      # access_key_id: AWS access key ID
      # secret_access_key: AWS secret access key
      # session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default
      profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when at-
    tempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style
    addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or

credentials Optional credentials shorthand for the config parameter
  • creds:
- **access_key_id**: AWS access key ID
- **secret_access_key**: AWS secret access key
- **session_token**: AWS temporary session token

  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

  **endpoint**  
  Optional shorthand for complete URL to use for the constructed client.

  **region**  
  Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- resourcegroupstaggingapi(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```
Operations

- **describe_report_creation**: Describes the status of the StartReportCreation operation
- **get_compliance_summary**: Returns a table that shows counts of resources that are noncompliant with their tag policies
- **get_resources**: Returns all the tagged or previously tagged resources that are located in the specified Amazon Web Services Region for the account
- **get_tag_keys**: Returns all tag keys currently in use in the specified Amazon Web Services Region for the calling account
- **get_tag_values**: Returns all tag values for the specified key that are used in the specified Amazon Web Services Region for the calling account
- **start_report_creation**: Generates a report that lists all tagged resources in the accounts across your organization and tells whether each resource is compliant with the effective tag policy
- **tag_resources**: Applies one or more tags to the specified resources
- **untag_resources**: Removes the specified tags from the specified resources

Examples

```r
## Not run:
svc <- resourcegroupstaggingapi()
svc$describe_report_creation(
  Foo = 123
)
## End(Not run)
```

---

**route53**

*Amazon Route 53*

**Description**

Amazon Route 53 is a highly available and scalable Domain Name System (DNS) web service. You can use Route 53 to:

- Register domain names.
  For more information, see [How domain registration works](#).
- Route internet traffic to the resources for your domain.
  For more information, see [How internet traffic is routed to your website or web application](#).
- Check the health of your resources.
  For more information, see [How Route 53 checks the health of your resources](#).

**Usage**

```r
route53(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```
Arguments

**config**
Optional configuration of credentials, endpoint, and/or region.

- **credentials**:  
  - **creds**:  
    - **access_key_id**: AWS access key ID  
    - **secret_access_key**: AWS secret access key  
    - **session_token**: AWS temporary session token  
  - **profile**: The name of a profile to use. If not given, then the default profile is used.  
  - **anonymous**: Set anonymous credentials.

- **endpoint**: The complete URL to use for the constructed client.

- **region**: The AWS Region used in instantiating the client.

- **close_connection**: Immediately close all HTTP connections.

- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.

- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [link](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials**
Optional credentials shorthand for the config parameter

- **creds**:  
  - **access_key_id**: AWS access key ID  
  - **secret_access_key**: AWS secret access key  
  - **session_token**: AWS temporary session token  

- **profile**: The name of a profile to use. If not given, then the default profile is used.

- **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- route53(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
    
```
secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
),
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

activate_key_signing_key
associate_vpc_with_hosted_zone
change_cidr_collection
change_resource_record_sets
change_tags_for_resource
create_cidr_collection
create_health_check
create_hosted_zone
create_key_signing_key
create_query_logging_config
create_reusable_delegation_set
create_traffic_policy
create_traffic_policy_instance
create_vpc_association_authorization
deactivate_key_signing_key
delete_cidr_collection
delete_health_check
delete_hosted_zone
delete_key_signing_key

Activates a key-signing key (KSK) so that it can be used for signing by DNSSEC

Associates an Amazon VPC with a private hosted zone

Creates, changes, or deletes CIDR blocks within a collection

Creates, changes, or deletes a resource record set, which contains authoritative DNS information

Adds, edits, or deletes tags for a health check or a hosted zone

Creates a CIDR collection in the current Amazon Web Services account

Creates a new health check

Creates a new public or private hosted zone

Creates a new key-signing key (KSK) associated with a hosted zone

Creates a configuration for DNS query logging

Creates a delegation set (a group of four name servers) that can be reused by multiple zones

Creates a traffic policy, which you use to create multiple DNS resource records

Creates resource record sets in a specified hosted zone based on the settings in a specified traffic policy

Authorizes the Amazon Web Services account that created a specified VPC to submit an AssociateVPCWithHostedZone request

Deactivates a key-signing key (KSK) so that it will not be used for signing by DNSSEC

Deletes a CIDR collection in the current Amazon Web Services account

Deletes a health check

Deletes a hosted zone

Deletes a key-signing key (KSK)
delete_query_logging_config
delete_reusable_delegation_set
delete_traffic_policy
delete_traffic_policy_instance
delete_vpc_association_authorization
disable_hosted_zone_dnssec
disassociate_vpc_from_hosted_zone
enable_hosted_zone_dnssec
get_account_limit
get_change
get_checker_ip_ranges
get_dnssec
get_geo_location
get_health_check
get_health_check_count
get_health_check_last_failure_reason
get_health_check_status
get_hosted_zone
get_hosted_zone_count
get_hosted_zone_limit
get_query_logging_config
get_reusable_delegation_set
get_reusable_delegation_set_limit
get_traffic_policy
get_traffic_policy_instance
get_traffic_policy_instance_count
list_cidr_blocks
list_cidr_collections
list_cidr_locations
list_geo_locations
list_health_checks
list_hosted_zones
list_hosted_zones_by_name
list_hosted_zones_by_vpc
list_query_logging_configs
list_resource_record_sets
list_reusable_delegation_sets
list_tags_for_resource
list_tags_for_resources
list_traffic_policies
list_traffic_policy_instances
list_traffic_policy_instances_by_hosted_zone
list_traffic_policy_instances_by_policy
list_traffic_policy_versions
list_vpc_association_authorizations
test_dns_answer
update_health_check
update_hosted_zone_comment

Delete a configuration for DNS query logging
Delete a reusable delegation set
Delete a traffic policy
Delete a traffic policy instance and all of the resource record sets that Amazon Route 53 created when you created the instance
Removes authorization to submit an AssociateVPCWithHostedZone request to associate a specified VPC with a hosted zone
Disables DNSSEC signing in a specific hosted zone
Disassociates an Amazon Virtual Private Cloud (Amazon VPC) from an Amazon Route 53 private hosted zone
Enables DNSSEC signing in a specific hosted zone
Gets the specified limit for the current account, for example, the maximum number of health checks
Returns the current status of a change batch request
Route 53 does not perform authorization for this API because it retrieves information that is already available to the public
Returns information about DNSSEC for a specific hosted zone, including the public keys that Amazon Route 53 uses to verify the zone's DNS traffic
Gets information about whether a specified geographic location is supported for Amazon Route 53 geolocation resource record sets
Gets information about a specified health check
Retrieves the number of health checks that are associated with the current Amazon Web Services account or a specific hosted zone
Gets the reason that a specified health check failed most recently
Gets status of a specified health check
Gets information about a specified hosted zone including the four name servers that are assigned to the hosted zone
Retrieves the number of hosted zones that are associated with the current Amazon Web Services account
Gets the specified limit for a specified hosted zone, for example, the maximum number of hosted zones that you can associate with the reusable delegation set
Gets information about a specified reusable delegation set, including the four name servers that are assigned to the delegation set
Gets the maximum number of hosted zones that you can associate with the specified reusable delegation set
Gets information about a specific traffic policy version
Gets information about a specified traffic policy instance
Gets the number of traffic policy instances that are associated with the current account
Returns a paginated list of location objects and their CIDR blocks
Returns a paginated list of CIDR collections in the Amazon Web Services account
Returns a paginated list of CIDR locations for the given collection (metadata only)
Retrieves a list of supported geographic locations
Retrieve a list of the health checks that are associated with the current Amazon Web Services account
Retrieves a list of the public and private hosted zones that are associated with the current Amazon Web Services account
Retrieves a list of your hosted zones in lexicographic order
Lists all the private hosted zones that a specified VPC is associated with, regardless of which Amazon Web Services account owns the VPC
Lists the configurations for DNS query logging that are associated with the current Amazon Web Services account
Lists the resource record sets in a specified hosted zone
Retrieves a list of the reusable delegation sets that are associated with the current Amazon Web Services account
Lists tags for one health check or hosted zone
Lists tags for up to 10 health checks or hosted zones
Gets information about the latest version for every traffic policy that is associated with the current account
Gets information about the traffic policy instances that you created by using the current account
Gets information about the traffic policy instances that you created in a specific hosted zone
Gets information about the traffic policy instances that you created by using a specific traffic policy
Gets information about all of the versions for a specified traffic policy
Gets a list of the VPCs that were created by other accounts and that can be associated with the specified hosted zone
Gets the value that Amazon Route 53 returns in response to a DNS request for a specified name and type
Updates an existing health check
Updates the comment for a specified hosted zone
update_traffic_policy_comment

Updates the comment for a specified traffic policy version

update_traffic_policy_instance

After you submit a UpdateTrafficPolicyInstance request, there’s a brief delay while Route 53 creates the resource record sets that are specified in the traffic policy definition.

Examples

```r
## Not run:
svc <- route53()
# The following example associates the VPC with ID vpc-1a2b3c4d with the
# hosted zone with ID Z3M3LMPEXAMPLE.
svc$associate_vpc_with_hosted_zone(
  Comment = "",
  HostedZoneId = "Z3M3LMPEXAMPLE",
  VPC = list(
    VPCId = "vpc-1a2b3c4d",
    VPCRegion = "us-east-2"
  )
)
## End(Not run)
```

route53domains

Amazon Route 53 Domains

Description

Amazon Route 53 API actions let you register domain names and perform related operations.

Usage

```r
route53domains(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

`config` Optional configuration of credentials, endpoint, and/or region.

- `credentials`:
  - `creds`:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

**credentials** Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- route53domains(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
  ),
  ...
)
```
timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
),
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

accept_domain_transfer_from_another_aws_account
associate_delegation_signer_to_domain
cancel_domain_transfer_to_another_aws_account
check_domain_availability
check_domain_transferability
delete_domain
delete_tags_for_domain
disable_domain_auto_renew
disable_domain_transfer_lock
disassociate_delegation_signer_from_domain
enable_domain_auto_renew
enable_domain_transfer_lock
get_contact_reachability_status
get_domain_detail
get_domain_suggestions
get_operation_detail
list_domains
list_operations
list_prices
list_tags_for_domain
push_domain
register_domain
reject_domain_transfer_from_another_aws_account
renew_domain
resend_contact_reachability_email
resend_operation_authorization
retrieve_domain_auth_code
transfer_domain
transfer_domain_to_another_aws_account

Accepts the transfer of a domain from another Amazon Web Services account to the current Amazon Web Services account.

Connects a delegation signer (DS) record to the domain name.

Cancels the transfer of a domain from the current Amazon Web Services account to another Amazon Web Services account.

Checks the availability of the specified domain name.

Checks whether a domain name can be transferred to Amazon Route 53.

Deletes a domain.

Deletes specified tags from a domain.

Disables automatic renewal of domain registration for a specified domain.

Removes the transfer lock on a domain (specifically the clientTransferProhibited status) to allow domain transfers.

Enables automatic renewal of domain registration for a specified domain.

Sets the transfer lock on the domain (specifically the clientTransferProhibited status) to prevent domain transfers.

Returns information about contact reachability status for a specified domain, or all domains.

Retrieves detailed information about a specified domain.

Retrieves a list of suggested domain names.

Returns the current status of an operation that is not completed.

Returns a list of domains that are registered with Amazon Route 53.

Lists the following prices for either all the TLDs supported by Route 53, or the specified TLD:

Retrieves the authorization code for a specified domain.

Transfers a domain from another registrar to Amazon Route 53.

Transfers a domain from another Amazon Web Services account to the current Amazon Web Services account.

For operations that require confirmation that the email address for the registrant contact has responded.

For operations that require confirmation that the email address for the registrant contact has responded.

For operations that require confirmation that the email address for the registrant contact has responded.

Resends the form of authorization email for this operation.

Retrieves the authorization code for the domain.

Transfers a domain from another registrar to Amazon Route 53.

Transfers a domain from the current Amazon Web Services account to another registrar.

For operations that require confirmation that the email address for the registrant contact has responded.

For operations that require confirmation that the email address for the registrant contact has responded.

For operations that require confirmation that the email address for the registrant contact has responded.
update_domain_contact
update_domain_contact_privacy
update_domain_nameservers
update_tags_for_domain
view_billing

**Examples**

```r
## Not run:
svc <- route53domains()
svc$accept_domain_transfer_from_another_aws_account(
  Foo = 123
)

## End(Not run)
```

---

**route53recoverycluster**

*Route53 Recovery Cluster*

**Description**

Welcome to the Routing Control (Recovery Cluster) API Reference Guide for Amazon Route 53 Application Recovery Controller.

With Route 53 ARC, you can use routing control with extreme reliability to recover applications by rerouting traffic across Availability Zones or Amazon Web Services Regions. Routing controls are simple on/off switches hosted on a highly available cluster in Route 53 ARC. A cluster provides a set of five redundant Regional endpoints against which you can run API calls to get or update the state of routing controls. To implement failover, you set one routing control to ON and another one to OFF, to reroute traffic from one Availability Zone or Amazon Web Services Region to another.

*Be aware that you must specify a Regional endpoint for a cluster when you work with API cluster operations to get or update routing control states in Route 53 ARC.* In addition, you must specify the US West (Oregon) Region for Route 53 ARC API calls. For example, use the parameter `--region us-west-2` with AWS CLI commands. For more information, see *Get and update routing control states using the API* in the *Amazon Route 53 Application Recovery Controller Developer Guide.*

This API guide includes information about the API operations for how to get and update routing control states in Route 53 ARC. To work with routing control in Route 53 ARC, you must first create the required components (clusters, control panels, and routing controls) using the recovery cluster configuration API.

For more information about working with routing control in Route 53 ARC, see the following:
• Create clusters, control panels, and routing controls by using API operations. For more information, see the Recovery Control Configuration API Reference Guide for Amazon Route 53 Application Recovery Controller.

• Learn about the components in recovery control, including clusters, routing controls, and control panels, and how to work with Route 53 ARC in the Amazon Web Services console. For more information, see Recovery control components in the Amazon Route 53 Application Recovery Controller Developer Guide.

• Route 53 ARC also provides readiness checks that continually audit resources to help make sure that your applications are scaled and ready to handle failover traffic. For more information about the related API operations, see the Recovery Readiness API Reference Guide for Amazon Route 53 Application Recovery Controller.

• For more information about creating resilient applications and preparing for recovery readiness with Route 53 ARC, see the Amazon Route 53 Application Recovery Controller Developer Guide.

Usage

route53recoverycluster(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

custom Optional configuration of credentials, endpoint, and/or region.

• config:
  • credentials:
    • creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    • profile: The name of a profile to use. If not given, then the default profile is used.
    • anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoint.html

credentials Optional credentials shorthand for the config parameter
- **creds**:  
  - `access_key_id`: AWS access key ID  
  - `secret_access_key`: AWS secret access key  
  - `session_token`: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint**  
Optional shorthand for complete URL to use for the constructed client.

**region**  
Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- route53recoverycluster(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```
Operations

get_routing_control_state Get the state for a routing control
list_routing_controls List routing control names and Amazon Resource Names (ARNs), as well as the routing control state for each routing control, along with the control panel name and control panel ARN for the routing controls
update_routing_control_state Set the state of the routing control to reroute traffic
update_routing_control_states Set multiple routing control states

Examples

```r
## Not run:
svc <- route53recoverycluster()
svc$get_routing_control_state(  
  Foo = 123
)

## End(Not run)
```

Description

Recovery Control Configuration API Reference for Amazon Route 53 Application Recovery Controller

Usage

```r
route53recoverycontrolconfig(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- config Optional configuration of credentials, endpoint, and/or region.
  - credentials:
    - creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
route53recoverycontrolconfig

- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

**credentials**
Optional credentials shorthand for the config parameter

  - **creds**:
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
    - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**
A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- route53recoverycontrolconfig(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
```
timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
),
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

create_cluster 
create_control_panel 
create_routing_control 
create_safety_rule 
delete_cluster 
delete_control_panel 
delete_routing_control 
delete_safety_rule 
describe_cluster 
describe_control_panel 
describe_routing_control 
describe_safety_rule 
get_resource_policy 
list_associated_route_53_health_checks 
list_clusters 
list_control_panels 
list_routing_controls 
list_safety_rules 
list_tags_for_resource 
tag_resource 
untag_resource 
update_control_panel 
update_routing_control 
update_safety_rule 

Create a new cluster
Creates a new control panel
Creates a new routing control
Creates a safety rule in a control panel
Delete a cluster
Deletes a control panel
Deletes a routing control
Deletes a safety rule
Display the details about a cluster
Displays details about a control panel
Displays details about a routing control
Returns information about a safety rule
Get information about the resource policy for a cluster
Returns an array of all Amazon Route 53 health checks associated with a specific routing control
Returns an array of all the clusters in an account
Returns an array of control panels in an account or in a cluster
Returns an array of routing controls for a control panel
List the safety rules (the assertion rules and gating rules) that you’ve defined for the routing controls in a control panel
Lists the tags for a resource
Adds a tag to a resource
Removes a tag from a resource
Updates a control panel
Updates a routing control
Update a safety rule (an assertion rule or gating rule)
route53recoveryreadiness

AWS Route53 Recovery Readiness

Description
Recovery readiness

Usage
route53recoveryreadiness(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)

Arguments
config Optional configuration of credentials, endpoint, and/or region.

- credentials:
  - creds:
    - access_key_id: AWS access key ID
    - secret_access_key: AWS secret access key
    - session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.
- endpoint: The complete URL to use for the constructed client.
- region: The AWS Region used in instantiating the client.
- close_connection: Immediately close all HTTP connections.
- timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

Examples
```r
## Not run:
svc <- route53recoverycontrolconfig()
svc$create_cluster(
  Foo = 123
)

## End(Not run)
```
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

  credentials  Optional credentials shorthand for the config parameter
  
  • **creds**:  
    – **access_key_id**: AWS access key ID  
    – **secret_access_key**: AWS secret access key  
    – **session_token**: AWS temporary session token  
  
  • **profile**: The name of a profile to use. If not given, then the default profile is used.
  
  • **anonymous**: Set anonymous credentials.

  endpoint  Optional shorthand for complete URL to use for the constructed client.

  region  Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- route53recoveryreadiness(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
```
anonymous = "logical",
endpoint = "string",
region = "string"
)

Operations

create_cell
create_cross_account_authorization
create_readiness_check
create_recovery_group
create_resource_set
delete_cell
delete_cross_account_authorization
delete_readiness_check
delete_recovery_group
delete_resource_set
get_architecture_recommendations
get_cell
get_cell_readiness_summary
get_readiness_check
get_readiness_check_resource_status
get_readiness_check_status
get_recovery_group
get_recovery_group_readiness_summary
get_resource_set
list_cells
list_cross_account_authorizations
list_readiness_checks
list_recovery_groups
list_resource_sets
list_rules
list_tags_for_resources
tag_resource
untag_resource
update_cell
update_readiness_check
update_recovery_group
update_resource_set

Examples

## Not run:
svc <- route53recoveryreadiness()
svc$create_cell(
Description

When you create a VPC using Amazon VPC, you automatically get DNS resolution within the VPC from Route 53 Resolver. By default, Resolver answers DNS queries for VPC domain names such as domain names for EC2 instances or Elastic Load Balancing load balancers. Resolver performs recursive lookups against public name servers for all other domain names.

You can also configure DNS resolution between your VPC and your network over a Direct Connect or VPN connection:

**Forward DNS queries from resolvers on your network to Route 53 Resolver**

DNS resolvers on your network can forward DNS queries to Resolver in a specified VPC. This allows your DNS resolvers to easily resolve domain names for Amazon Web Services resources such as EC2 instances or records in a Route 53 private hosted zone. For more information, see How DNS Resolvers on Your Network Forward DNS Queries to Route 53 Resolver in the Amazon Route 53 Developer Guide.

**Conditionally forward queries from a VPC to resolvers on your network**

You can configure Resolver to forward queries that it receives from EC2 instances in your VPCs to DNS resolvers on your network. To forward selected queries, you create Resolver rules that specify the domain names for the DNS queries that you want to forward (such as example.com), and the IP addresses of the DNS resolvers on your network that you want to forward the queries to. If a query matches multiple rules (example.com, acme.example.com), Resolver chooses the rule with the most specific match (acme.example.com) and forwards the query to the IP addresses that you specified in that rule. For more information, see How Route 53 Resolver Forwards DNS Queries from Your VPCs to Your Network in the Amazon Route 53 Developer Guide.

Like Amazon VPC, Resolver is Regional. In each Region where you have VPCs, you can choose whether to forward queries from your VPCs to your network (outbound queries), from your network to your VPCs (inbound queries), or both.

Usage

```r
route53resolver(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

## End(Not run)
Arguments

`config` Optional configuration of credentials, endpoint, and/or region.

- **credentials**:
  - `creds`:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - `profile`: The name of a profile to use. If not given, then the default profile is used.
  - `anonymous`: Set anonymous credentials.
- `endpoint`: The complete URL to use for the constructed client.
- `region`: The AWS Region used in instantiating the client.
- `close_connection`: Immediately close all HTTP connections.
- `timeout`: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- `s3_force_path_style`: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.

`credentials` Optional credentials shorthand for the `config` parameter

- `creds`:
  - `access_key_id`: AWS access key ID
  - `secret_access_key`: AWS secret access key
  - `session_token`: AWS temporary session token
- `profile`: The name of a profile to use. If not given, then the default profile is used.
- `anonymous`: Set anonymous credentials.

`endpoint` Optional shorthand for complete URL to use for the constructed client.

`region` Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- route53resolver(
    config = list(
        credentials = list(
            creds = list(
                access_key_id = "string",
                secret_access_key = "string",
                session_token = "string",
                profile = "string",
                anonymous = TRUE
            )
        ),
        endpoint = "string",
        region = "string"
    )
)```
secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
),
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

**Operations**

- **associate_firewall_rule_group**: Associates a FirewallRuleGroup with a VPC, to provide DNS filtering for the VPC.
- **associate_resolver_endpoint_ip_address**: Adds IP addresses to an inbound or an outbound Resolver endpoint.
- **associate_resolver_query_log_config**: Associates an Amazon VPC with a specified query logging configuration.
- **associate_resolver_rule**: Associates a Resolver rule with a VPC.
- **create_firewall_domain_list**: Creates an empty firewall domain list for use in DNS Firewall rules.
- **create_firewall_rule**: Creates a single DNS Firewall rule in the specified rule group, using the specified domain list.
- **create_firewall_rule_group**: Creates an empty DNS Firewall rule group for filtering DNS network traffic in a VPC.
- **create_outpost_resolver**: Creates a Route 53 Resolver on an Outpost.
- **create_resolver_endpoint**: Creates a Resolver endpoint.
- **create_resolver_query_log_config**: Creates a Resolver query logging configuration, which defines where you want Resolver to save DNS query logs that originate in your VPCs.
- **create_resolver_rule**: For DNS queries that originate in your VPCs, specifies which Resolver endpoint the queries pass through, one domain name that you want to forward to your network, and the IP addresses of the DNS resolvers in your network.
- **delete_firewall_domain_list**: Deletes the specified domain list.
- **delete_firewall_rule**: Deletes the specified firewall rule.
- **delete_firewall_rule_group**: Deletes the specified firewall rule group.
- **delete_outpost_resolver**: Deletes a Resolver on the Outpost.
- **delete_resolver_endpoint**: Deletes a Resolver endpoint.
- **delete_resolver_query_log_config**: Deletes a query logging configuration.
- **delete_resolver_rule**: Deletes a Resolver rule.
- **disassociate_firewall_rule_group**: Disassociates a FirewallRuleGroup from a VPC, to remove DNS filtering from the VPC.
- **disassociate_resolver_endpoint_ip_address**: Removes IP addresses from an inbound or an outbound Resolver endpoint.
disassociate_resolver_query_log_config
disassociate_resolver_rule
get_firewall_config
get_firewall_domain_list
get_firewall_rule_group
get_firewall_rule_group_association
get_firewall_rule_log_config
get_firewall_rule_log_config_association
get_firewall_rule_log_config_policy
get_resolver_endpoint
get_outpost_resolver
get_resolver_config
get_resolver_dnssec_config
get_resolver_query_log_config
get_resolver_query_log_config_association
get_resolver_query_log_config_policy
get_resolver_rule
get_resolver_rule_association
get_resolver_rule_policy
import_firewall_domains
list_firewall_configs
list_firewall_domain_lists
list_firewall_domains
list_firewall_rule_group_associations
list_firewall_rule_groups
list_firewall_rules
list_outpost_resolvers
list_resolver_configs
list_resolver_dnssec_configs
list_resolver_endpoint_ip_addresses
list_resolver_endpoints
list_resolver_query_log_config_associations
list_resolver_query_log_configs
list_resolver_rule_associations
list_resolver_rules
list_tags_for_resource
put_firewall_rule_group_policy
put_resolver_query_log_config_policy
put_resolver_rule_policy
tag_resource
untag_resource
update_firewall_config
update_firewall_domain_list
update_firewall_rule
update_firewall_rule_group_association
update_outpost_resolver
update_resolver_config
update_resolver_dnssec_config
update_resolver_endpoint
update_resolver_rule
Disassociates a VPC from a query logging configuration
Removes the association between a specified Resolver rule and a specified VPC
Retrieves the configuration of the firewall behavior provided by DNS Firewall
Retrieves the specified firewall domain list
Retrieves the firewall rule group association, which enables DNS filtering for a VPC
Returns the Identity and Access Management (Amazon Web Services IAM) policy for a firewall rule group
Gets information about a specified Resolver on the Outpost, such as its instance count and type
Retrieves the behavior configuration of Route 53 Resolver behavior for a single VPC
Gets DNSSEC validation information for a specified resource
Gets information about a specified Resolver endpoint, such as whether it’s an inbound or outbound endpoint
Gets information about a specified Resolver query logging configuration, such as the location where logs are sent
Gets information about a specified association between a Resolver query logging configuration and a VPC
Gets information about a query logging policy
Gets information about a specified Resolver rule, such as the domain name that the rule forwards DNS queries for
Gets information about an association between a specified Resolver rule and a specified VPC
Gets information about the Resolver rule policy for a specified rule
Imports domain names from a file into a domain list, for use in a DNS firewall configuration
Retrieves the firewall configurations that you have defined
Retrieves the firewall domain lists that you have defined
Retrieves the domains that you have defined for the specified firewall domain list
Retrieves the firewall rule group associations that you have defined
Retrieves the minimal high-level information for the rule groups that you have defined
Retrieves the firewall rules that you have defined for the specified firewall rule group
Lists all the Resolvers on Outposts that were created using the current Amazon Web Services (AWS) account
Retrieves the Resolver configurations that you have defined
Lists the configurations for DNSSEC validation that are associated with the current AWS account
Gets the IP addresses for a specified Resolver endpoint
Lists all the Resolver endpoints that were created using the current Amazon Web Services account
Lists information about associations between Amazon VPCs and query logging configurations
Lists information about the specified query logging configurations
Lists the associations that were created between Resolver rules and VPCs using the current AWS account
Lists the Resolver rules that were created using the current Amazon Web Services account
Lists the tags that you associated with the specified resource
Attaches an Identity and Access Management (Amazon Web Services IAM) policy to a specified resource
Specifies an Amazon Web Services account that you want to share a query logging configuration
Specifies an Amazon Web Services rule that you want to share with another account
Adds one or more tags to a specified resource
Removes one or more tags from a specified resource
Updates the configuration of the firewall behavior provided by DNS Firewall for a VPC
Updates the firewall domain list from an array of domain specifications
Updates the specified firewall rule
Changes the association of a FirewallRuleGroup with a VPC
You can use UpdateOutpostResolver to update the instance count, type, or name of a specified Outpost Resolver
Updates the behavior configuration of Route 53 Resolver behavior for a single VPC
Updates an existing DNSSEC validation configuration
Updates the name, or endpoint type for an inbound or an outbound Resolver endpoint
Updates settings for a specified Resolver rule
Examples

```r
## Not run:
svc <- route53resolver()
svc$associate_firewall_rule_group(
  Foo = 123
)

## End(Not run)
```

## s3

*Amazon Simple Storage Service*

**Description**

Amazon Simple Storage Service

**Usage**

```r
s3(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

**Arguments**

- `config` Optional configuration of credentials, endpoint, and/or region.
  - `credentials`: 
    - `creds`:
      * `access_key_id`: AWS access key ID
      * `secret_access_key`: AWS secret access key
      * `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
    - `anonymous`: Set anonymous credentials.
  - `endpoint`: The complete URL to use for the constructed client.
  - `region`: The AWS Region used in instantiating the client.
  - `close_connection`: Immediately close all HTTP connections.
  - `timeout`: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - `s3_force_path_style`: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
- `credentials` Optional credentials shorthand for the config parameter
- **creds:**
  - `access_key_id`: AWS access key ID
  - `secret_access_key`: AWS secret access key
  - `session_token`: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

### Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

### Service syntax

```r
svc <- s3(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```
Operations

abort_multipart_upload
complete_multipart_upload
copy_object
create_bucket
create_multipart_upload
create_session
delete_bucket
delete_bucket_analytics_configuration
delete_bucket_cors
delete_bucket_encryption
delete_bucket_intelligent_tiering_configuration
delete_bucket_inventory_configuration
delete_bucket_lifecycle
delete_bucket_metrics_configuration
delete_bucket_ownership_controls
delete_bucket_policy
delete_bucket_replication
delete_bucket_tagging
delete_bucket_website
delete_object
delete_objects
delete_object_tagging
delete_public_access_block
download_file
generate_presigned_url
generate_presigned_url
get_bucket_accelerate_configuration
get_bucket_acl
get_bucket_analytics_configuration
get_bucket_cors
get_bucket_encryption
get_bucket_intelligent_tiering_configuration
get_bucket_inventory_configuration
get_bucket_lifecycle
get_bucket_location
get_bucket_logging
get_bucket_metrics_configuration
get_bucket_notification
get_bucket_notification_configuration
get_bucket_ownership_controls
get_bucket_policy
get_bucket_policy_status
get_bucket_replication
get_bucket_request_payment
get_bucket_tagging
get_bucket_versioning

This operation aborts a multipart upload
Completes a multipart upload by assembling previously uploaded parts
Creates a copy of an object that is already stored in Amazon S3
This action creates an Amazon S3 bucket
This action initiates a multipart upload and returns an upload ID
Creates a session that establishes temporary security credentials to support fast authentication and authorization for the Zonal endpoint APIs on directory buckets
Deletes the S3 bucket
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
Deletes the policy of a specified bucket
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
Removes an object from a bucket
This operation enables you to delete multiple objects from a bucket using a single HTTP request
This operation is not supported by directory buckets
This operation is not supported by directory buckets
Download a file from S3 and store it at a specified file location
@title Generate a presigned url given a client, its method, and arguments
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
For an updated version of this API, see GetBucketLifecycleConfiguration
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
Returns the policy of a specified bucket
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
get_bucket_website
get_object
get_object_acl
get_object_attributes
get_object_legal_hold
get_object_lock_configuration
get_object_retention
get_object_tagging
get_object_torrent
get_public_access_block
head_bucket
head_object
list_bucket_analytics_configurations
list_bucket_intelligent_tiering_configurations
list_bucket_inventory_configurations
list_bucket_metrics_configurations
list_buckets
list_directory_buckets
list_multipart_uploads
list_objects
list_objects_v2
list_object_versions
list_parts
put_bucket_accelerate_configuration
put_bucket_acl
put_bucket_analytics_configuration
put_bucket_cors
put_bucket_encryption
put_bucket_intelligent_tiering_configuration
put_bucket_inventory_configuration
put_bucket_lifecycle
put_bucket_lifecycle_configuration
put_bucket_logging
put_bucket_metrics_configuration
put_bucket_notification
put_bucket_notification_configuration
put_bucket_ownership_controls
put_bucket_policy
put_bucket_replication
put_bucket_request_payment
put_bucket_tagging
put_bucket_versioning
put_bucket_website
put_object
put_object_acl
put_object_legal_hold
put_object_lock_configuration
put_object_retention

This operation is not supported by directory buckets
Retrieves an object from Amazon S3
This operation is not supported by directory buckets
Retrieves all the metadata from an object without returning the object itself
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
You can use this operation to determine if a bucket exists and if you have permission to access it
The HEAD operation retrieves metadata from an object without returning the object itself
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
Returns a list of all Amazon S3 directory buckets owned by the authenticated sender of the request
This operation lists in-progress multipart uploads in a bucket
This operation is not supported by directory buckets
Returns some or all (up to 1,000) of the objects in a bucket with each request
This operation is not supported by directory buckets
Lists the parts that have been uploaded for a specific multipart upload
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
Applies an Amazon S3 bucket policy to an Amazon S3 bucket
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
Adds an object to a bucket
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
s3control

Description

Amazon Web Services S3 Control provides access to Amazon S3 control plane actions.

Usage

s3control(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)

Arguments

config            Optional configuration of credentials, endpoint, and/or region.

  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key

Examples

## Not run:
svc <- s3()
# The following example aborts a multipart upload.
svc$abort_multipart_upload(
  Bucket = "examplebucket",
  Key = "bigobject",
  UploadId = "xadcOB_7YPB0JuoFiQ9cz4P3Pe6F1z04f7wN93uHsNBEw97p15eNwzExg0LA"
)
## End(Not run)
* session_token: AWS temporary session token
  – profile: The name of a profile to use. If not given, then the default profile is used.
  – anonymous: Set anonymous credentials.

• endpoint: The complete URL to use for the constructed client.

• region: The AWS Region used in instantiating the client.

• close_connection: Immediately close all HTTP connections.

• timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

• s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

• sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html

credentials Optional credentials shorthand for the config parameter

  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token

  • profile: The name of a profile to use. If not given, then the default profile is used.

  • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- s3control(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string"
  )
)
```
close_connection = "logical",
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

associate_access_grants_identity_center
create_access_grant
create_access_grants_instance
create_access_grants_location
create_access_point
create_access_point_for_object_lambda
create_bucket
create_job
create_multi_region_access_point
create_storage_lens_group
delete_access_grant
delete_access_grants_instance
delete_access_grants_instance_resource_policy
delete_access_grants_location
delete_access_point
delete_access_point_for_object_lambda
delete_access_point_policy
delete_access_point_policy_for_object_lambda
delete_bucket
delete_bucket_lifecycle_configuration
delete_bucket_policy
delete_bucket_replication
delete_bucket_tagging
delete_job_tagging
delete_multi_region_access_point
delete_public_access_block
delete_storage_lens_configuration
delete_storage_lens_configuration_tagging
	note: Associate your S3 Access Grants instance with an Amazon Web Services IAM Identity Center instance
	note: Creates an access grant that gives a grantee access to your S3 data
	note: Creates an S3 Access Grants instance, which serves as a logical grouping for access grants
	note: The S3 data location that you would like to register in your S3 Access Grants instance
	note: This operation is not supported by directory buckets
	note: This operation is not supported by directory buckets
	note: This action creates an Amazon S3 on Outposts bucket
	note: This operation creates an S3 Batch Operations job
	note: This operation is not supported by directory buckets
	note: Creates a new S3 Storage Lens group and associates it with the specified Amazon Web Services account ID
	note: Deletes the access grant from the S3 Access Grants instance
	note: Deletes your S3 Access Grants instance
	note: Deletes the resource policy of the S3 Access Grants instance
	note: Deregisters a location from your S3 Access Grants instance
	note: This operation is not supported by directory buckets
	note: This operation is not supported by directory buckets
	note: This operation is not supported by directory buckets
	note: This action deletes an Amazon S3 on Outposts bucket
	note: This action deletes an Amazon S3 on Outposts bucket’s lifecycle configuration
	note: This action deletes an Amazon S3 on Outposts bucket policy
	note: This operation deletes an Amazon S3 on Outposts bucket’s replication configuration
	note: This action deletes an Amazon S3 on Outposts bucket’s tags
	note: Removes the entire tag set from the specified S3 Batch Operations job
	note: This operation is not supported by directory buckets
	note: This operation is not supported by directory buckets
	note: This operation is not supported by directory buckets
	note: This operation is not supported by directory buckets
delete_storage_lens_group
describe_job
describe_multi_region_access_point_operation
dissociate_access_grants_identity_center
get_access_grant
get_access_grants_instance
get_access_grants_instance_for_prefix
get_access_grants_instance_resource_policy
get_access_grants_location
get_access_point
get_access_point_configuration_for_object_lambda
get_access_point_for_object_lambda
get_access_point_policy
get_access_point_policy_for_object_lambda
get_access_point_policy_status
get_access_point_policy_status_for_object_lambda
get_bucket
get_bucket_lifecycle_configuration
get_bucket_policy
get_bucket_replication
get_bucket_tagging
get_bucket_versioning
get_data_access
get_job_tagging
get_multi_region_access_point
get_multi_region_access_point_policy
get_multi_region_access_point_policy_status
get_multi_region_access_point_routes
get_public_access_block
get_storage_lens_configuration
get_storage_lens_configuration_tagging
get_storage_lens_group
list_access_grants
list_access_grants_instances
list_access_grants_locations
list_access_points
list_access_points_for_object_lambda
list_jobs
list_multi_region_access_points
list_regional_buckets
list_storage_lens_configurations
list_storage_lens_groups
list_tags_for_resource
put_access_grants_instance_resource_policy
put_access_point_configuration_for_object_lambda
put_access_point_policy
put_access_point_policy_for_object_lambda
put_bucket_lifecycle_configuration

Deletes an existing S3 Storage Lens group
Retrieves the configuration parameters and status for a Batch Operations job
This operation is not supported by directory buckets
Dissociates the Amazon Web Services IAM Identity Center instance from an S3 Access Grants instance
Get the details of an access grant from your S3 Access Grants instance
Retrieves the S3 Access Grants instance for a Region in your account
Retrieve the S3 Access Grants instance that contains a particular prefix
Retrieves the details of a particular location registered in your S3 Access Grants instance
This operation is not supported by directory buckets
This operation is not supported by directory buckets
Get an Amazon S3 on Outposts bucket
This action gets an Amazon S3 on Outposts bucket’s lifecycle configuration
This action gets a policy for an Amazon S3 on Outposts bucket
This operation gets an Amazon S3 on Outposts bucket’s replication configuration
This action gets an Amazon S3 on Outposts bucket’s tags
This operation returns the versioning state for S3 on Outposts buckets or objects
Returns a temporary access credential from S3 Access Grants to the grantee or client application
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation allows you to list all the Amazon Web Services resources
Updates the resource policy of the S3 Access Grants instance
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation puts a lifecycle configuration to an Amazon S3 on Outposts bucket
**Description**

Amazon S3 on Outposts provides access to S3 on Outposts operations.

**Usage**

```r
s3outposts(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

This action puts a bucket policy to an Amazon S3 on Outposts bucket
This action creates an Amazon S3 on Outposts bucket’s replication configuration
This action puts tags on an Amazon S3 on Outposts bucket
This operation sets the versioning state for S3 on Outposts buckets only
Sets the supplied tag-set on an S3 Batch Operations job
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
This operation is not supported by directory buckets
Creates a new Amazon Web Services resource tag or updates an existing resource tag
This operation removes the specified Amazon Web Services resource tag
Updates the IAM role of a registered location in your S3 Access Grants instance
Updates an existing S3 Batch Operations job’s priority
Updates the status for the specified job
Updates the existing Storage Lens group
Arguments

config Optional configuration of credentials, endpoint, and/or region.
  - credentials:
    - creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    - profile: The name of a profile to use. If not given, then the default profile is used.
    - anonymous: Set anonymous credentials.
  - endpoint: The complete URL to use for the constructed client.
  - region: The AWS Region used in instantiating the client.
  - close_connection: Immediately close all HTTP connections.
  - timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  - sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy

credentials Optional credentials shorthand for the config parameter
  - creds:
    - access_key_id: AWS access key ID
    - secret_access_key: AWS secret access key
    - session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- s3outposts(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      )
    )
  )
)```
secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
),
    credentials = list(
        creds = list(
            access_key_id = "string",
            secret_access_key = "string",
            session_token = "string"
        ),
        profile = "string",
        anonymous = "logical"
    ),
    endpoint = "string",
    region = "string"
)

**Operations**

- **create_endpoint**  Creates an endpoint and associates it with the specified Outpost
- **delete_endpoint**  Deletes an endpoint
- **list_endpoints**   Lists endpoints associated with the specified Outpost
- **list_outposts_with_s3**  Lists the Outposts with S3 on Outposts capacity for your Amazon Web Services account
- **list_shared_endpoints**  Lists all endpoints associated with an Outpost that has been shared by Amazon Web Services Resource Access Manager (RAM)

**Examples**

```r
## Not run:
svc <- s3outposts()
svc$create_endpoint(
    Foo = 123
)

## End(Not run)
```
Amazon SageMaker Service

Description

Provides APIs for creating and managing SageMaker resources.

Other Resources:

- SageMaker Developer Guide
- Amazon Augmented AI Runtime API Reference

Usage

sagemaker(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:  
    - **creds**:  
      - *access_key_id*: AWS access key ID
      - *secret_access_key*: AWS secret access key
      - *session_token*: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:  
    - *access_key_id*: AWS access key ID
- **secret_access_key**: AWS secret access key
- **session_token**: AWS temporary session token

  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**
A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- sagemaker(
    config = list(
        credentials = list(
            creds = list(
                access_key_id = "string",
                secret_access_key = "string",
                session_token = "string"
            ),
            profile = "string",
            anonymous = "logical"
        ),
        endpoint = "string",
        region = "string",
        close_connection = "logical",
        timeout = "numeric",
        s3_force_path_style = "logical",
        sts_regional_endpoint = "string"
    ),
    credentials = list(
        creds = list(
            access_key_id = "string",
            secret_access_key = "string",
            session_token = "string"
        ),
        profile = "string",
        anonymous = "logical"
    ),
    endpoint = "string",
    region = "string"
)
```
Operations
add_association
add_tags
associate_trial_component
batch_describe_model_package
create_action
create_algorithm
create_app
create_app_image_config
create_artifact
create_auto_ml_job
create_auto_ml_job_v2
create_cluster
create_code_repository
create_compilation_job
create_context
create_data_quality_job_definition
create_device_fleet
create_domain
create_edge_deployment_plan
create_edge_deployment_stage
create_edge_packaging_job
create_endpoint
create_endpoint_config
create_experiment
create_feature_group
create_flow_definition
create_hub
create_human_task_ui
create_hyper_parameter_tuning_job
create_image
create_image_version
create_inference_component
create_inference_experiment
create_inference_recommendations_job
create_labeling_job
create_model
create_model_bias_job_definition
create_model_card
create_model_card_export_job
create_model_explainability_job_definition
create_model_package
create_model_package_group
create_model_quality_job_definition
create_monitoring_schedule
create_notebook_instance
create_notebook_instance_lifecycle_config
create_pipeline
create_presigned_domain_url

Creates an association between the source and the destination
Adds or overwrites one or more tags for the specified SageMaker resource
Associates a trial component with a trial
This action batch describes a list of versioned model packages
Creates an action
Create a machine learning algorithm that you can use in SageMaker and list in the Amazon Web Services Marketplace
Creates a running app for the specified UserProfile
Creates a configuration for running a SageMaker image as a KernelGateway app
Creates an artifact
Creates an Autopilot job also referred to as Autopilot experiment or AutoML job
Creates an Autopilot job also referred to as Autopilot experiment or Autopilot job V2
Creates a SageMaker HyperPod cluster
Creates a Git repository as a resource in your SageMaker account
Starts a model compilation job
Creates a context
Creates a definition for a job that monitors data quality and drift
Creates a device fleet
Creates a Domain
Creates an edge deployment plan, consisting of multiple stages
Creates a new stage in an existing edge deployment plan
Starts a SageMaker Edge Manager model packaging job
Creates an endpoint using the endpoint configuration specified in the request
Creates an endpoint configuration that SageMaker hosting services uses
Creates a SageMaker experiment
Create a new FeatureGroup
Creates a flow definition
Create a hub
Defines the settings you will use for the human review workflow user interface
Starts a hyperparameter tuning job
Creates a custom SageMaker image
Creates a version of the SageMaker image specified by ImageName
Creates an inference component, which is a SageMaker hosting object that accepts an input dataset and returns an output dataset
Creates an inference experiment using the configurations specified in the request
Starts a recommendation job
Creates a job that uses workers to label the data objects in your input dataset
Creates a model in SageMaker
Creates the definition for a model bias job
Creates an Amazon SageMaker Model Card
Creates an Amazon SageMaker Model Card export job
Creates the definition for a model explainability job
Creates a model package that you can use to create SageMaker models or list on Amazon Web Services Marketplace
Creates a model group
Creates a definition for a job that monitors model quality and drift
Creates a schedule that regularly starts Amazon SageMaker Processing jobs
Creates an Amazon SageMaker notebook instance
Creates a lifecycle configuration that you can associate with a notebook
Creates a pipeline using a JSON pipeline definition
Creates a URL for a specified UserProfile in a Domain
create_presigned_notebook_instance_url
create_processing_job
create_project
create_space
create_studio_lifecycle_config
create_training_job
create_transform_job
create_trial
create_trial_component
create_user_profile
create_workforce
create_workteam
delete_action
delete_algorithm
delete_app
delete_app_image_config
delete_artifact
delete_association
delete_cluster
delete_code_repository
delete_compilation_job
delete_context
delete_data_quality_job_definition
delete_device_fleet
delete_domain
delete_edge_deployment_plan
delete_edge_deployment_stage
delete_endpoint
delete_endpoint_config
delete_experiment
delete_feature_group
delete_flow_definition
delete_hub
delete_hub_content
delete_human_task_ui
delete_hyper_parameter_tuning_job
delete_image
delete_image_version
delete_inference_component
delete_inference_experiment
delete_model
delete_model_bias_job_definition
delete_model_card
delete_model_explainability_job_definition
delete_model_package
delete_model_package_group
delete_model_package_group_policy
delete_model_quality_job_definition

Returns a URL that you can use to connect to the Jupyter server from a notebook instance.

Creates a processing job.

Creates a machine learning (ML) project that can contain one or more templates that set up an ML pipeline from training to deploying an approved model.

Creates a space used for real time collaboration in a domain.

Creates a new Amazon SageMaker Studio Lifecycle Configuration.

Starts a model training job.

Starts a transform job.

Creates an Amazon SageMaker trial.

Creates a trial component, which is a stage of a machine learning trial.

Creates a user profile.

Use this operation to create a workforce.

Creates a new work team for labeling your data.

Deletes an action.

Removes the specified algorithm from your account.

Used to stop and delete an app.

Deletes an AppImageConfig.

Deletes an artifact.

Deletes an association.

Delete a SageMaker HyperPod cluster.

Deletes the specified Git repository from your account.

Deletes the specified compilation job.

Deletes an context.

Deletes a data quality monitoring job definition.

Deletes a fleet.

Used to delete a domain.

Deletes an edge deployment plan if (and only if) all the stages in the plan are inactive.

Delete a stage in an edge deployment plan if (and only if) the stage is inactive.

Deletes an endpoint.

Deletes an endpoint configuration.

Deletes an SageMaker experiment.

Delete the FeatureGroup and any data that was written to the OnlineStore.

Deletes the specified flow definition.

Delete a hub.

Delete the contents of a hub.

Use this operation to delete a human task user interface (worker task template).

Deletes a hyperparameter tuning job.

Deletes a SageMaker image and all versions of the image.

Deletes a version of a SageMaker image.

Deletes an inference component.

Deletes an inference experiment.

Deletes a model.

Deletes an Amazon SageMaker model bias job definition.

Deletes an Amazon SageMaker Model Card.

Deletes an Amazon SageMaker model explainability job definition.

Deletes a model package.

Deletes the specified model group.

Deletes a model group resource policy.

Deletes the specified model quality monitoring job definition.
delete_monitoring_schedule
delete_notebook_instance
delete_notebook_instance_lifecycle_config
delete_pipeline
delete_project
delete_space
delete_studio_lifecycle_config
delete_tags
delete_trial
delete_trial_component
delete_user_profile
delete_workforce
delete_workteam
deregister_devices
describe_action
describe_algorithm
describe_app
describe_app_image_config
describe_artifact
describe_auto_ml_job
describe_auto_ml_job_v2
describe_cluster
describe_cluster_node
describe_code_repository
describe_compilation_job
describe_context
describe_data_quality_job_definition
describe_device
describe_device_fleet
describe_domain
describe_edge_deployment_plan
describe_edge_packaging_job
describe_endpoint
describe_endpoint_config
describe_experiment
describe_feature_group
describe_feature_metadata
describe_flow_definition
describe_hub
describe_hub_content
describe_human_task_ui
describe_hyper_parameter_tuning_job
describe_image
describe_image_version
describe_inference_component
describe_inference_experiment
describe_inference_recommendations_job
describe_labeling_job

Deletes a monitoring schedule
Deletes an SageMaker notebook instance
Deletes a notebook instance lifecycle configuration
Deletes a pipeline if there are no running instances of the pipeline
Delete the specified project
Used to delete a space
Deletes the Amazon SageMaker Studio Lifecycle Configuration
Deletes the specified tags from an SageMaker resource
Deletes the specified trial
Deletes the specified trial component
Deletes a user profile
Use this operation to delete a workforce
Deletes an existing work team
Deregisters the specified devices
Describes an action
Returns a description of the specified algorithm that is in your account
Describes the app
Describes an AppImageConfig
Describes an artifact
Returns information about an AutoML job created by calling CreateAutoMLJob
Returns information about an AutoML job created by calling CreateAutoMLJobV2 or CreateAutoMLJob
Retrieves information of a SageMaker HyperPod cluster
Retrieves information of an instance (also called a node interchangeably)
Gets details about the specified Git repository
Returns information about a model compilation job
Describes a context
Gets the details of a data quality monitoring job definition
Describes the device
A description of the fleet the device belongs to
The description of the domain
Describes an edge deployment plan with deployment status per stage
A description of edge packaging jobs
Returns the description of an endpoint
Returns the description of an endpoint configuration created using the CreateEndpointConfig API
Provides a list of an experiment’s properties
Use this operation to describe a FeatureGroup
Shows the metadata for a feature within a feature group
Returns information about the specified flow definition
Describe a hub
Describe the content of a hub
Returns information about the requested human task user interface (worker)
Returns a description of a hyperparameter tuning job, depending on the stage
Describes a SageMaker image
Describes a version of a SageMaker image
Returns information about an inference component
Returns details about an inference experiment
Provides the results of the Inference Recommender job
Gets information about a labeling job
describe_lineage_group
describe_model
describe_model_bias_job_definition
describe_model_card
describe_model_card_export_job
describe_model_explainability_job_definition
describe_model_package
describe_model_package_group
describe_model_quality_job_definition
describe_monitoring_schedule
describe_notebook_instance
describe_notebook_instance_lifecycle_config
describe_pipeline
describe_pipeline_definition_for_execution
describe_pipeline_execution
describe_processing_job
describe_project
describe_space
describe_studio_lifecycle_config
describe_subscribed_workteam
describe_training_job
describe_transform_job
describe_trial
describe_trial_component
describe_user_profile
describe_workforce
describe_workteam
disable_sagemaker_servicecatalog_portfolio
disassociate_trial_component
enable_sagemaker_servicecatalog_portfolio
get_device_fleet_report
get_lineage_group_policy
get_model_package_group_policy
get_sagemaker_servicecatalog_portfolio_status
get_scaling_configuration_recommendation
get_search_suggestions
import_hub_content
list_actions
list_algorithms
list_aliases
list_app_image_configs
list_apps
list_artifacts
list_associations
list_auto_ml_jobs
list_candidates_for_auto_ml_job
list_cluster_nodes
list_clusters

Provides a list of properties for the requested lineage group
Describes a model that you created using the CreateModel API
Returns a description of a model bias job definition
Describes the content, creation time, and security configuration of an Amazon SageMaker Model Card
Returns a description of a model explainability job definition
Returns a description of the specified model package, which is used to create SageMaker models
Gets a description for the specified model group
Returns a description of a model quality job definition
Describes the schedule for a monitoring job
Returns information about a notebook instance
Returns a description of a notebook instance lifecycle configuration
Describes the details of a pipeline
Describes the details of an execution’s pipeline definition
Describes the details of a pipeline execution
Returns a description of a processing job
Describes the details of a project
Describes the space
Describes the Amazon SageMaker Studio Lifecycle Configuration
Gets information about a work team provided by a vendor
Returns information about a training job
Returns information about a transform job
Provides a list of a trial’s properties
Provides a list of a trials component’s properties
Describes a user profile
Lists private workforce information, including workforce name, Amazon SageMaker ARN, and allowed IP address ranges
Disables using Service Catalog in SageMaker
Disassociates a trial component from a trial
Enables using Service Catalog in SageMaker
Describes a fleet
The resource policy for the lineage group
Gets a resource policy that manages access for a model group
Gets the status of Service Catalog in SageMaker
Starts an Amazon SageMaker Inference Recommender autoscaling recommendation job
An auto-complete API for the search functionality in the SageMaker console
Import hub content
Lists the actions in your account and their properties
Lists the machine learning algorithms that have been created
Lists the aliases of a specified image or image version
Lists the AppImageConfigs in your account and their properties
Lists apps
Lists the artifacts in your account and their properties
Lists the associations in your account and their properties
Request a list of jobs
Lists the candidates created for the job
Retrieves the list of instances (also called nodes interchangeably) in a SageMaker HyperPod cluster
Retrieves the list of SageMaker HyperPod clusters
- list_code_repositories
- list_compilation_jobs
- list_contexts
- list_data_quality_job_definitions
- list_device_fleets
- list_devices
- list_domains
- list_edge_deployment_plans
- list_edge_packaging_jobs
- list_endpoint_configs
- list_endpoints
- list_experiments
- list_feature_groups
- list_flow_definitions
- list_hub_contents
- list_hub_content_versions
- list_hubs
- list_human_task_uis
- list_hyper_parameter_tuning_jobs
- list_images
- list_image_versions
- list_inference_components
- list_inference_experiments
- list_inference_recommendations_jobs
- list_inference_recommendations_job_steps
- list_labeling_jobs
- list_labeling_jobs_for_workteam
- list_lineage_groups
- list_model_bias_job_definitions
- list_model_card_export_jobs
- list_model_cards
- list_model_card_versions
- list_model_explainability_job_definitions
- list_model_metadata
- list_model_package_groups
- list_model_packages
- list_model_quality_job_definitions
- list_models
- list_monitoring_alert_history
- list_monitoring_alerts
- list_monitoring_executions
- list_monitoring_schedules
- list_notebook_instance.lifecycle_configs
- list_notebook_instances
- list_pipeline_executions
- list_pipeline_execution_steps
- list_pipeline_parameters_for_execution
- list_pipelines
- list_code_repositories
- list_compilation_jobs
- list_contexts
- list_data_quality_job_definitions
- list_device_fleets
- list_devices
- list_domains
- list_edge_deployment_plans
- list_edge_packaging_jobs
- list_endpoint_configs
- list_endpoints
- list_experiments
- list_feature_groups
- list_flow_definitions
- list_hub_contents
- list_hub_content_versions
- list_hubs
- list_human_task_uis
- list_hyper_parameter_tuning_jobs
- list_images
- list_image_versions
- list_inference_components
- list_inference_experiments
- list_inference_recommendations_jobs
- list_inference_recommendations_job_steps
- list_labeling_jobs
- list_labeling_jobs_for_workteam
- list_lineage_groups
- list_model_bias_job_definitions
- list_model_card_export_jobs
- list_model_cards
- list_model_card_versions
- list_model_explainability_job_definitions
- list_model_metadata
- list_model_package_groups
- list_model_packages
- list_model_quality_job_definitions
- list_models
- list_monitoring_alert_history
- list_monitoring_alerts
- list_monitoring_executions
- list_monitoring_schedules
- list_notebook_instance.lifecycle_configs
- list_notebook_instances
- list_pipeline_executions
- list_pipeline_execution_steps
- list_pipeline_parameters_for_execution
- list_pipelines
- Gets a list of the Git repositories in your account
- Lists model compilation jobs that satisfy various filters
- Lists the contexts in your account and their properties
- Lists the data quality job definitions in your account
- Returns a list of devices in the fleet
- A list of devices
- Lists the domains
- Lists all edge deployment plans
- Returns a list of edge packaging jobs
- Lists endpoint configurations
- Lists endpoints
- Lists all the experiments in your account
- List FeatureGroups based on given filter and order
- Returns information about the flow definitions in your account
- List the contents of a hub
- List hub content versions
- List all existing hubs
- Returns information about the human task user interfaces in your account
- Gets a list of HyperParameterTuningJobSummary objects that describe
- Lists the images in your account and their properties
- Lists the versions of a specified image and their properties
- Lists the inference components in your account and their properties
- Returns the list of all inference experiments
- Lists recommendation jobs that satisfy various filters
- Returns a list of the subtasks for an Inference Recommender job
- Gets a list of labeling jobs
- Gets a list of labeling jobs assigned to a specified work team
- A list of lineage groups shared with your Amazon Web Services account
- Lists model bias jobs definitions that satisfy various filters
- List the export jobs for the Amazon SageMaker Model Card
- List existing model cards
- List existing versions of an Amazon SageMaker Model Card
- Lists model explainability job definitions that satisfy various filters
- Lists the domain, framework, task, and model name of standard machine
- Gets a list of the model groups in your Amazon Web Services account
- Lists the model packages that have been created
- Gets a list of model quality monitoring job definitions in your account
- Lists models created with the CreateModel API
- Gets a list of past alerts in a model monitoring schedule
- Gets the alerts for a single monitoring schedule
- Returns list of all monitoring job executions
- Returns list of all monitoring schedules
- Lists notebook instance lifestyle configurations created with the Create
- Returns a list of the SageMaker notebook instances in the requester’s ac
- Gets a list of the pipeline executions
- Gets a list of PipeLineExecutionStep objects
- Gets a list of parameters for a pipeline execution
- Gets a list of pipelines
list_processing_jobs
list_projects
list_resource_catalogs
list_spaces
list_stage_devices
list_studio_lifecycle_configs
list_subscribed_workteams
list_tags
list_training_jobs
list_training_jobs_for_hyper_parameter_tuning_job
list_transform_jobs
list_trial_components
list_trials
list_user_profiles
list_workforces
list_workteams
put_model_package_group_policy
query_lineage
register_devices
render_ui_template
retry_pipeline_execution
search
send_pipeline_execution_step_failure
send_pipeline_execution_step_success
start_edge_deployment_stage
start_inference_experiment
start_monitoring_schedule
start_notebook_instance
start_pipeline_execution
stop_auto_ml_job
stop_compilation_job
stop_edge_deployment_stage
stop_edge_packaging_job
stop_hyper_parameter_tuning_job
stop_inference_experiment
stop_inference_recommendations_job
stop_labeling_job
stop_monitoring_schedule
stop_notebook_instance
stop_pipeline_execution
stop_processing_job
stop_training_job
stop_transform_job
update_action
update_app_image_config
update_artifact
update_cluster
update_cluster_software

Lists processing jobs that satisfy various filters
Gets a list of the projects in an Amazon Web Services account
Lists Amazon SageMaker Catalogs based on given filters and orders
Lists spaces
Lists devices allocated to the stage, containing detailed device information
Lists the Amazon SageMaker Studio Lifecycle Configurations in your AWS account
Gets a list of the work teams that you are subscribed to in the Amazon Web Services Marketplace
Returns the tags for the specified SageMaker resource
Lists training jobs
Gets a list of TrainingJobSummary objects that describe the training jobs
Lists transform jobs
Lists the trial components in your account
Lists the trials in your account
Lists user profiles
Use this operation to list all private and vendor workforces in an Amazon Web Services Region
Gets a list of private work teams that you have defined in a region
Adds a resource policy to control access to a model group
Use this operation to inspect your lineage and discover relationships between data items
Register devices
Renders the UI template so that you can preview the worker’s experience
Retry the execution of the pipeline
Finds SageMaker resources that match a search query
Notifies the pipeline that the execution of a callback step failed, along with a message describing why
Notifies the pipeline that the execution of a callback step succeeded and provides a list of the step’s output parameters
Starts a stage in an edge deployment plan
Starts an inference experiment
Starts a previously stopped monitoring schedule
Launches an ML compute instance with the latest version of the libraries
Starts a pipeline execution
A method for forcing a running job to shut down
Stops a model compilation job
Stops a stage in an edge deployment plan
Request to stop an edge packaging job
Stops a running hyperparameter tuning job and all running training jobs
Starts an inference experiment
Stops an Inference Recommender job
Stops a running labeling job
Stops a previously started monitoring schedule
Terminates the ML compute instance
Stops a pipeline execution
Stops a processing job
Stops a training job
Stops a batch transform job
Updates an action
Updates the properties of an AppImageConfig
Updates an artifact
Updates a SageMaker HyperPod cluster
Updates the platform software of a SageMaker HyperPod cluster for security patches
update_code_repository
update_context
update_device_fleet
update_devices
update_domain
update_endpoint
update_endpoint_weights_and_capacities
update_experiment
update_feature_group
update_feature_metadata
update_hub
update_image
update_image_version
update_inference_component
update_inference_component_runtime_config
update_inference_experiment
update_model_card
update_model_package
update_monitoring_alert
update_monitoring_schedule
update_notebook_instance
update_notebook_instance_lifecycle_config
update_pipeline
update_pipeline_execution
update_project
update_space
update_training_job
update_trial
update_trial_component
update_user_profile
update_workforce
update_workteam

Updates the specified Git repository with the specified values
Updates a context
Updates a fleet of devices
Updates one or more devices in a fleet
Updates the default settings for new user profiles in the domain
Deploys the EndpointConfig specified in the request to a new fleet of instances
Updates variant weight of one or more variants associated with an existing endpoint
Adds, updates, or removes the description of an experiment
Updates the feature group by either adding features or updating the online store
Updates the description and parameters of the feature group
Updates a hub
Updates the properties of a SageMaker image
Updates the properties of a SageMaker image version
Updates an inference component
Runtime settings for a model that is deployed with an inference component
Updates an inference experiment that you created
Update an Amazon SageMaker Model Card
Updates a versioned model
Update the parameters of a model monitor alert
Updates a previously created schedule
Updates a notebook instance
Updates a notebook instance lifecycle configuration created with the CreateNotebookInstanceLifecycleConfig API
Updates a pipeline
Updates a pipeline execution
Updates a machine learning (ML) project that is created from a template
Updates the settings of a space
Update a model training job to request a new Debugger profiling configuration
Updates the display name of a trial
Updates one or more properties of a trial component
Updates a user profile
Use this operation to update your workforce
Updates an existing work team with new member definitions or descriptions

Examples

```r
## Not run:
svc <- sagemaker()
svc$add_association(
    Foo = 123
)

## End(Not run)
```
Amazon Sagemaker Edge Manager

Description

SageMaker Edge Manager dataplane service for communicating with active agents.

Usage

sagemakeredgemanager(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
    - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
• **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**
A client for the service. You can call the service’s operations using syntax like `svc(operation(...))`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- sagemakeredgemanager(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string",
      ),
      profile = "string",
      anonymous = "logical",
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string",
    ),
    profile = "string",
    anonymous = "logical",
  ),
  endpoint = "string",
  region = "string"
)
```

**Operations**

- **get_deployments**
  Use to get the active deployments from a device

- **get_device_registration**
  Use to check if a device is registered with SageMaker Edge Manager

- **send_heartbeat**
  Use to get the current status of devices registered on SageMaker Edge Manager
Examples

```r
## Not run:
svc <- sagemakeredgemanager()
svc$get_deployments(
  Foo = 123
)
## End(Not run)
```

---

sagemakerfeaturestore

---

Amazon SageMaker Feature Store Runtime

Description

Contains all data plane API operations and data types for the Amazon SageMaker Feature Store. Use this API to put, delete, and retrieve (get) features from a feature store.

Use the following operations to configure your **OnlineStore** and **OfflineStore** features, and to create and manage feature groups:

- `CreateFeatureGroup`
- `DeleteFeatureGroup`
- `DescribeFeatureGroup`
- `ListFeatureGroups`

Usage

```r
sagemakerfeaturestore(config = list(),
                      credentials = list(),
                      endpoint = NULL,
                      region = NULL)
```

Arguments

- `config`: Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
    - `creds`:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
– **profile**: The name of a profile to use. If not given, then the default profile is used.
– **anonymous**: Set anonymous credentials.
  • **endpoint**: The complete URL to use for the constructed client.
  • **region**: The AWS Region used in instantiating the client.
  • **close_connection**: Immediately close all HTTP connections.
  • **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
  • **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials**

Optional credentials shorthand for the config parameter

• **creds**:
  – **access_key_id**: AWS access key ID
  – **secret_access_key**: AWS secret access key
  – **session_token**: AWS temporary session token
  • **profile**: The name of a profile to use. If not given, then the default profile is used.
  • **anonymous**: Set anonymous credentials.

**endpoint**

Optional shorthand for complete URL to use for the constructed client.

**region**

Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- sagemakerfeaturestoreruntime(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
```
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
  creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
  ),
  profile = "string",
  anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)

Operations

- **batch_get_record**: Retrieves a batch of Records from a FeatureGroup
- **delete_record**: Deletes a Record from a FeatureGroup in the OnlineStore
- **get_record**: Use for OnlineStore serving from a FeatureStore
- **put_record**: The PutRecord API is used to ingest a list of Records into your feature group

Examples

```r
## Not run:
svc <- sagemakerfeaturestoreruntime()
svc$batch_get_record(
  Foo = 123
)
## End(Not run)
```

---

**Amazon SageMaker geospatial capabilities**

**Description**

Provides APIs for creating and managing SageMaker geospatial resources.
Usage

```r
sagemakergeospatialcapabilities(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **creds**:
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
    - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.
- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
    - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

- **endpoint**: Optional shorthand for complete URL to use for the constructed client.
- **region**: Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

svc <- sagemakergeospatialcapabilities(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)

Operations

del**ete**_earth_observation_job Use this operation to delete an Earth Observation job
delete_vector_enrichment_job Use this operation to delete a Vector Enrichment job
export_earth_observation_job Use this operation to export results of an Earth Observation job and optionally source images used as input to the EOJ to an Amazon S3 location
export_vector_enrichment_job Use this operation to copy results of a Vector Enrichment job to an Amazon S3 location
get_earth_observation_job Get the details for a previously initiated Earth Observation job
get_raster_data_collection Gets a web mercator tile for the given Earth Observation job
get_vector_enrichment_job Use this operation to get details of a specific raster data collection
get_tile Gets a web mercator tile for the given Earth Observation job
list_earth_observation_jobs Use this operation to get a list of the Earth Observation jobs associated with the calling Amazon Web Services account
list_raster_data_collections Retrieves details of a Vector Enrichment Job for a given job Amazon Resource Name (ARN)
list_tags_for_resource Use this operation to get raster data collections
list_vector_enrichment_jobs Lists the tags attached to the resource
search_raster_data_collection Retrieves a list of vector enrichment jobs
start_earth_observation_job Allows you run image query on a specific raster data collection to get a list of the satellite images
start_earth_observation_job Use this operation to create an Earth observation job
sagemakermetrics

- start_vector_enrichment_job: Creates a Vector Enrichment job for the supplied job type
- stop_earth_observation_job: Use this operation to stop an existing earth observation job
- stop_vector_enrichment_job: Stops the Vector Enrichment job for a given job ARN
- tag_resource: The resource you want to tag
- untag_resource: The resource you want to untag

Examples

```r
## Not run:
svc <- sagemakergeospatialcapabilities()
svc$delete_earth_observation_job(
  Foo = 123
)
## End(Not run)
```

---

**Description**

Contains all data plane API operations and data types for Amazon SageMaker Metrics. Use these APIs to put and retrieve (get) features related to your training run.

- batch_put_metrics

**Usage**

```r
sagemakermetrics(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

**Arguments**

- `config`: Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
    - `creds`:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
– **profile**: The name of a profile to use. If not given, then the default profile is used.
– **anonymous**: Set anonymous credentials.

**endpoint**: The complete URL to use for the constructed client.
**region**: The AWS Region used in instantiating the client.
**close_connection**: Immediately close all HTTP connections.
**timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
**s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
**sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

**credentials** Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- sagemakermetrics(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    ...)
)```
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

batch_put_metrics Used to ingest training metrics into SageMaker

Examples

## Not run:
svc <- sagemakermetrics()
svc$batch_put_metrics(
    Foo = 123
)

## End(Not run)

---

sagemakerruntime Amazon SageMaker Runtime

Description

The Amazon SageMaker runtime API.

Usage

sagemakerruntime(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
Arguments

config  Optional configuration of credentials, endpoint, and/or region.

  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com-sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials  Optional credentials shorthand for the config parameter

  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

endpoint  Optional shorthand for complete URL to use for the constructed client.

region  Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- sagemakerruntime(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
      
```
Operations

invokes_endpoint

After you deploy a model into production using Amazon SageMaker hosting services, your client applications use this API to get inferences from the model hosted at the specified endpoint.

invokes_endpoint_async

After you deploy a model into production using Amazon SageMaker hosting services, your client applications use this API to get inferences from the model hosted at the specified endpoint in an asynchronous manner.

invokes_endpoint_with_response_stream

Invokes a model at the specified endpoint to return the inference response as a stream.

Examples

```r
## Not run:
svc <- sagemakerruntime()
svc$invokes_endpoint(
  Foo = 123
)

## End(Not run)
```
AWS Savings Plans

Description

Savings Plans are a pricing model that offer significant savings on Amazon Web Services usage (for example, on Amazon EC2 instances). You commit to a consistent amount of usage per hour, in the specified currency, for a term of one or three years, and receive a lower price for that usage. For more information, see the Amazon Web Services Savings Plans User Guide.

Usage

```r
savingsplans(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **creds**:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - `profile`: The name of a profile to use. If not given, then the default profile is used.
  - `anonymous`: Set anonymous credentials.
  - `endpoint`: The complete URL to use for the constructed client.
  - `region`: The AWS Region used in instantiating the client.
  - `close_connection`: Immediately close all HTTP connections.
  - `timeout`: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - `s3_force_path_style`: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  - `sts_regional_endpoint`: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
- `session_token`: AWS temporary session token
  - `profile`: The name of a profile to use. If not given, then the default profile is used.
  - `anonymous`: Set anonymous credentials.

endpoint  Optional shorthand for complete URL to use for the constructed client.
region    Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- savingsplans(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations
create_savings_plan
delete_queued_savings_plan
describe_savings_plan_rates
describe_savings_plans
describe_savings_plans_offering_rates
describe_savings_plans_offerings
list_tags_for_resource
return_savings_plan
tag_resource
untag_resource

create_savings_plan
Creates a Savings Plan
delete_queued_savings_plan
Deletes the queued purchase for the specified Savings Plan
describe_savings_plan_rates
Describes the rates for the specified Savings Plan
describe_savings_plans
Describes the specified Savings Plans
describe_savings_plans_offering_rates
Describes the offering rates for the specified Savings Plans
describe_savings_plans_offerings
Describes the offerings for the specified Savings Plans
list_tags_for_resource
Lists the tags for the specified resource
return_savings_plan
Returns the specified Savings Plan
tag_resource
Adds the specified tags to the specified resource
untag_resource
Removes the specified tags from the specified resource

Examples

```r
## Not run:
svc <- savingsplans()
svc$create_savings_plan(
  Foo = 123
)
## End(Not run)
```

schemas
Schemas

Description
Amazon EventBridge Schema Registry

Usage

```r
schemas(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

cfg Optional configuration of credentials, endpoint, and/or region.

- **credentials:**
  - creds:
    - **access_key_id:** AWS access key ID
    - **secret_access_key:** AWS secret access key
    - **session_token:** AWS temporary session token
  - **profile:** The name of a profile to use. If not given, then the default profile is used.
  - **anonymous:** Set anonymous credentials.
• **endpoint**: The complete URL to use for the constructed client.
• **region**: The AWS Region used in instantiating the client.
• **close_connection**: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [link](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-episode.html)

**credentials** Optional credentials shorthand for the config parameter

• **creds**:  
  – **access_key_id**: AWS access key ID  
  – **secret_access_key**: AWS secret access key  
  – **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- schemas(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  )
)```

credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
endpoint = "string",
region = "string"
)

**Operations**

<table>
<thead>
<tr>
<th>Operation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>create_discoverer</td>
<td>Creates a discoverer</td>
</tr>
<tr>
<td>create_registry</td>
<td>Creates a registry</td>
</tr>
<tr>
<td>create_schema</td>
<td>Creates a schema definition</td>
</tr>
<tr>
<td>delete_discoverer</td>
<td>Deletes a discoverer</td>
</tr>
<tr>
<td>delete_registry</td>
<td>Deletes a Registry</td>
</tr>
<tr>
<td>delete_resource_policy</td>
<td>Delete the resource-based policy attached to the specified registry</td>
</tr>
<tr>
<td>delete_schema</td>
<td>Delete a schema definition</td>
</tr>
<tr>
<td>delete_schema_version</td>
<td>Delete the schema version definition</td>
</tr>
<tr>
<td>describe_code_binding</td>
<td>Describe the code binding URI</td>
</tr>
<tr>
<td>describe_discoverer</td>
<td>Describes the discoverer</td>
</tr>
<tr>
<td>describe_registry</td>
<td>Describes the registry</td>
</tr>
<tr>
<td>describe_schema</td>
<td>Retrieve the schema definition</td>
</tr>
<tr>
<td>export_schema</td>
<td>Export schema</td>
</tr>
<tr>
<td>get_code_binding_source</td>
<td>Get the code binding source URI</td>
</tr>
<tr>
<td>get_discovered_schema</td>
<td>Get the discovered schema that was generated based on sampled events</td>
</tr>
<tr>
<td>get_resource_policy</td>
<td>Retrieves the resource-based policy attached to a given registry</td>
</tr>
<tr>
<td>list_discoverers</td>
<td>List the discoverers</td>
</tr>
<tr>
<td>list_registries</td>
<td>List the registries</td>
</tr>
<tr>
<td>list_schemas</td>
<td>List the schemas</td>
</tr>
<tr>
<td>list_schema_versions</td>
<td>Provides a list of the schema versions and related information</td>
</tr>
<tr>
<td>list_tags_for_resource</td>
<td>Get tags for resource</td>
</tr>
<tr>
<td>put_code_binding</td>
<td>Put code binding URI</td>
</tr>
<tr>
<td>put_resource_policy</td>
<td>The name of the policy</td>
</tr>
<tr>
<td>search_schemas</td>
<td>Search the schemas</td>
</tr>
<tr>
<td>start_discoverer</td>
<td>Starts the discoverer</td>
</tr>
<tr>
<td>stop_discoverer</td>
<td>Stops the discoverer</td>
</tr>
<tr>
<td>tag_resource</td>
<td>Add tags to a resource</td>
</tr>
<tr>
<td>untag_resource</td>
<td>Removes tags from a resource</td>
</tr>
<tr>
<td>update_discoverer</td>
<td>Updates the discoverer</td>
</tr>
<tr>
<td>update_registry</td>
<td>Updates a registry</td>
</tr>
<tr>
<td>update_schema</td>
<td>Updates the schema definition</td>
</tr>
</tbody>
</table>
Examples

```r
## Not run:
svc <- schemas()
svc$create_discoverer(
   Foo = 123
)

## End(Not run)
```

---

**secretsmanager**  
**AWS Secrets Manager**

**Description**

Amazon Web Services Secrets Manager

Amazon Web Services Secrets Manager provides a service to enable you to store, manage, and retrieve, secrets.

This guide provides descriptions of the Secrets Manager API. For more information about using this service, see the Amazon Web Services Secrets Manager User Guide.

**API Version**

This version of the Secrets Manager API Reference documents the Secrets Manager API version 2017-10-17.

For a list of endpoints, see Amazon Web Services Secrets Manager endpoints.

**Support and Feedback for Amazon Web Services Secrets Manager**

We welcome your feedback. Send your comments to awssecretsmanager-feedback@amazon.com, or post your feedback and questions in the Amazon Web Services Secrets Manager Discussion Forum. For more information about the Amazon Web Services Discussion Forums, see Forums Help.

**Logging API Requests**

Amazon Web Services Secrets Manager supports Amazon Web Services CloudTrail, a service that records Amazon Web Services API calls for your Amazon Web Services account and delivers log files to an Amazon S3 bucket. By using information that’s collected by Amazon Web Services CloudTrail, you can determine the requests successfully made to Secrets Manager, who made the request, when it was made, and so on. For more about Amazon Web Services Secrets Manager and support for Amazon Web Services CloudTrail, see Logging Amazon Web Services Secrets Manager Events with Amazon Web Services CloudTrail in the Amazon Web Services Secrets Manager User Guide. To learn more about CloudTrail, including enabling it and find your log files, see the Amazon Web Services CloudTrail User Guide.
Usage

```python
secretsmanager(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **creds**:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.
- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

- **endpoint**: Optional shorthand for complete URL to use for the constructed client.
- **region**: Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```r
svc <- secretsmanager(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `batch_get_secret_value`: Retrieves the contents of the encrypted fields SecretString or SecretBinary for up to 20 secrets.
- `cancel_rotate_secret`: Turns off automatic rotation, and if a rotation is currently in progress, cancels the rotation.
- `create_secret`: Creates a new secret.
- `delete_resource_policy`: Deletes the resource-based permission policy attached to the secret.
- `delete_secret`: Deletes a secret and all of its versions.
- `describe_secret`: Retrieves the details of a secret.
- `get_random_password`: Generates a random password.
- `get_resource_policy`: Retrieves the JSON text of the resource-based policy document attached to the secret.
- `get_secret_value`: Retrieves the contents of the encrypted fields SecretString or SecretBinary from the specified version of a secret, whichever contains content.
- `list_secrets`: Lists the secrets that are stored by Secrets Manager in the Amazon Web Services account.
- `list_secret_version_ids`: Lists the versions of a secret.
- `put_resource_policy`: Attaches a resource-based permission policy to a secret.
- `put_secret_value`: Creates a new version with a new encrypted secret value and attaches it to the secret.
- `remove_regions_from_replication`: For a secret that is replicated to other Regions, deletes the secret replicas from the Region.
replicate_secret_to_regions  Replicates the secret to a new Regions
restore_secret           Cancels the scheduled deletion of a secret by removing the DeletedDate time stamp
rotate_secret           Configures and starts the asynchronous process of rotating the secret
stop_replication_to_replica  Removes the link between the replica secret and the primary secret and promotes the replica to a primary secret
tag_resource             Attaches tags to a secret
untag_resource           Removes specific tags from a secret
update_secret            Modifies the details of a secret, including metadata and the secret value
update_secret_version_stage  Modifies the staging labels attached to a version of a secret
validate_resource_policy Validates that a resource policy does not grant a wide range of principals access to your secret.

Examples

```r
## Not run:
svc <- secretsmanager()

# The following example gets the values for three secrets.
svc$batch_get_secret_value(
  SecretIdList = list(
    "MySecret1",
    "MySecret2",
    "MySecret3"
  )
)

## End(Not run)

securityhub  AWS SecurityHub

Description

Security Hub provides you with a comprehensive view of your security state in Amazon Web Services and helps you assess your Amazon Web Services environment against security industry standards and best practices.

Security Hub collects security data across Amazon Web Services accounts, Amazon Web Services, and supported third-party products and helps you analyze your security trends and identify the highest priority security issues.

To help you manage the security state of your organization, Security Hub supports multiple security standards. These include the Amazon Web Services Foundational Security Best Practices (FSBP) standard developed by Amazon Web Services, and external compliance frameworks such as the Center for Internet Security (CIS), the Payment Card Industry Data Security Standard (PCI DSS), and the National Institute of Standards and Technology (NIST). Each standard includes several security controls, each of which represents a security best practice. Security Hub runs checks against security controls and generates control findings to help you assess your compliance against security best practices.
In addition to generating control findings, Security Hub also receives findings from other Amazon Web Services, such as Amazon GuardDuty and Amazon Inspector, and supported third-party products. This gives you a single pane of glass into a variety of security-related issues. You can also send Security Hub findings to other Amazon Web Services and supported third-party products.

Security Hub offers automation features that help you triage and remediate security issues. For example, you can use automation rules to automatically update critical findings when a security check fails. You can also leverage the integration with Amazon EventBridge to trigger automatic responses to specific findings.

This guide, the Security Hub API Reference, provides information about the Security Hub API. This includes supported resources, HTTP methods, parameters, and schemas. If you’re new to Security Hub, you might find it helpful to also review the Security Hub User Guide. The user guide explains key concepts and provides procedures that demonstrate how to use Security Hub features. It also provides information about topics such as integrating Security Hub with other Amazon Web Services.

In addition to interacting with Security Hub by making calls to the Security Hub API, you can use a current version of an Amazon Web Services command line tool or SDK. Amazon Web Services provides tools and SDKs that consist of libraries and sample code for various languages and platforms, such as PowerShell, Java, Go, Python, C++, and .NET. These tools and SDKs provide convenient, programmatic access to Security Hub and other Amazon Web Services. They also handle tasks such as signing requests, managing errors, and retrying requests automatically. For information about installing and using the Amazon Web Services tools and SDKs, see Tools to Build on Amazon Web Services.

With the exception of operations that are related to central configuration, Security Hub API requests are executed only in the Amazon Web Services Region that is currently active or in the specific Amazon Web Services Region that you specify in your request. Any configuration or settings change that results from the operation is applied only to that Region. To make the same change in other Regions, call the same API operation in each Region in which you want to apply the change. When you use central configuration, API requests for enabling Security Hub, standards, and controls are executed in the home Region and all linked Regions. For a list of central configuration operations, see the Central configuration terms and concepts section of the Security Hub User Guide.

The following throttling limits apply to Security Hub API operations.

- batch_enable_standards - RateLimit of 1 request per second. BurstLimit of 1 request per second.
- get_findings - RateLimit of 3 requests per second. BurstLimit of 6 requests per second.
- batch_import_findings - RateLimit of 10 requests per second. BurstLimit of 30 requests per second.
- batch_update_findings - RateLimit of 10 requests per second. BurstLimit of 30 requests per second.
- update_standards_control - RateLimit of 1 request per second. BurstLimit of 5 requests per second.
- All other operations - RateLimit of 10 requests per second. BurstLimit of 30 requests per second.
Usage

securityhub(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

config  Optional configuration of credentials, endpoint, and/or region.

- credentials:
  - creds:
    - access_key_id: AWS access key ID
    - secret_access_key: AWS secret access key
    - session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.

- endpoint: The complete URL to use for the constructed client.

- region: The AWS Region used in instantiating the client.

- close_connection: Immediately close all HTTP connections.

- timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

- s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

- sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials  Optional credentials shorthand for the config parameter

- creds:
  - access_key_id: AWS access key ID
  - secret_access_key: AWS secret access key
  - session_token: AWS temporary session token

- profile: The name of a profile to use. If not given, then the default profile is used.

- anonymous: Set anonymous credentials.

endpoint  Optional shorthand for complete URL to use for the constructed client.

region  Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```
svc <- securityhub(
  config = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string",
  close_connection = "logical",
  timeout = "numeric",
  s3_force_path_style = "logical",
  sts_regional_endpoint = "string"
),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `accept_administrator_invitation` Accepts the invitation to be a member account and be monitored by the Security Hub administrator account that the invitation was sent from
- `accept_invitation` This method is deprecated
- `batch_delete_automation_rules` Deletes one or more automation rules
- `batch_disable_standards` Disables the standards specified by the provided StandardsSubscriptionArns
- `batch_enable_standards` Enables the standards specified by the provided StandardsArn
- `batch_get_automation_rules` Returns a list of details for automation rules based on rule Amazon Resource Names (ARNs)
- `batch_get_configuration_policy_associations` Returns associations between an Security Hub configuration and a batch of target accounts, organizational units, or the root
- `batch_get_security_controls` Provides details about a batch of security controls for the current Amazon Web Services account and Amazon Web Services Region
- `batch_get_standards_control_associations` For a batch of security controls and standards, identifies whether each control is currently enabled or disabled in a standard
- `batch_import_findings` Imports security findings generated by a finding provider into Security Hub
- `batch_update_automation_rules` Updates one or more automation rules based on rule Amazon Resource Names (ARNs)
- `batch_update_configuration_policy_associations` Used by Security Hub customers to update information about their investigation
- `batch_update_findings` For a batch of security controls and standards, this operation updates the enablement status of a control in a standard
- `batch_update_standards_control_associations` For a batch of security controls and standards, this operation updates the enablement status of a control in a standard
- `create_action_target` Creates a custom action target in Security Hub
create_automation_rule
create_configuration_policy
create_finding_aggregator
create_insight
decline_invitations
delete_action_target
delete_configuration_policy
delete_finding_aggregator
delete_insight
delete_invitations
delete_members
describe_action_targets
describe_hub
describe_organization_configuration
describe_products
describe_standards
describe_standards_controls
disable_import_findings_for_product
disable_organization_admin_account
disable_security_hub
disassociate_from_administrator_account
disassociate_from_master_account
disassociate_members
enable_import_findings_for_product
enable_organization_admin_account
enable_security_hub
get_administrator_account
get_configuration_policy
get_configuration_policy_association
get_enabled_standards
get_finding_aggregator
get_finding_history
get_findings
get_insight_results
get_insights
get_invitations_count
get_master_account
get_members
get_security_control_definition
invite_members
list_automation_rules
list_configuration_policies
list_configuration_policy_associations
list_enabled_products_for_import
list_finding_aggregators
list_invitations
list_members

create_automation_rule
Creates an automation rule based on input parameters
create_configuration_policy
Creates a configuration policy with the defined configuration
Used to enable finding aggregation
create_finding_aggregator
Creates a custom insight in Security Hub
create_insight
Creates a member association in Security Hub between the specified accounts and the account used to make the request, which is the administrator account
decline_invitations
Declines invitations to become a member account
delete_action_target
Deletes a custom action target from Security Hub
delete_configuration_policy
Deletes a configuration policy
delete_finding_aggregator
Deletes a finding aggregator
delete_insight
Deletes the insight specified by the InsightArn
delete_invitations
Deletes invitations received by the Amazon Web Services account to become a member account
delete_members
Deletes the specified member accounts from Security Hub
describe_action_targets
Returns a list of the custom action targets in Security Hub in your account
describe_hub
Returns details about the Hub resource in your account, including the HubArn and the time when you enabled Security Hub
describe_organization_configuration
Returns information about the way your organization is configured in Security Hub
describe_products
Returns information about product integrations in Security Hub
describe_standards
Returns a list of the available standards in Security Hub
disassociate_from_administrator_account
Disassociates the current Security Hub member account from the associated administrator account
disable_import_findings_for_product
Disables the integration of the specified product with Security Hub
disable_organization_admin_account
Disables a Security Hub administrator account in your account only in the current Amazon Web Services Region
Disassociate_from_master_account
Disassociates the current Security Hub member account from the associated administrator account
This method is deprecated
disassociate_members
Disassociates the specified member accounts from the associated administrator account
delete_import_findings_for_product
Enables the integration of a partner product with Security Hub
delete_organization_admin_account
Designates the Security Hub administrator account for an organization
delete_security_hub
Enables Security Hub for your account in the current Region or the Region you specify in the request
describe_configuration_policy
Provides the details for the Security Hub administrator account for the current Region
get_configuration_policy
Provides information about a configuration policy
get_enabled_standards
Returns a list of the standards that are currently enabled
get_finding_aggregator
Returns the current finding aggregation configuration
get_finding_history
Returns history for a Security Hub finding in the last 90 days
get_findings
Returns a list of findings that match the specified criteria
get_automaton_rules
Lists the results of the Security Hub insight specified by the insight ARN
get_automaton_rules
Lists and describes insights for the specified insight ARNs
get_automaton_rules
Returns the count of all Security Hub membership invitations that were sent to the current member account
get_automaton_rules
This method is deprecated
get_automaton_rules
Retrieves the definition of a security control
get_automaton_rules
Invites other Amazon Web Services accounts to become member accounts for your Security Hub
get_automaton_rules
A list of automation rules and their metadata for the calling account
get_automaton_rules
Lists the configuration policies that the Security Hub delegated administrator owns
get_automaton_rules
Provides information about the associations for your configuration policies and any enabled products
get_automaton_rules
Lists all findings-generating solutions (products) that you are subscribed to receive findings from in Security Hub
get_automaton_rules
If finding aggregation is enabled, then ListFindingAggregators returns the ARN of the finding aggregator that is currently enabled
get_automaton_rules
Lists all Security Hub membership invitations that were sent to the current Amazon Web Services account
get_automaton_rules
Lists details about all member accounts for the current Security Hub administrator
get_automaton_rules
Examples

```r
## Not run:
svc <- securityhub()
# The following example demonstrates how an account can accept an
# invitation from the Security Hub administrator account to be a member
# account. This operation is applicable only to member accounts that are
# not added through AWS Organizations.
svc$accept_administrator_invitation(
  AdministratorId = "123456789012",
  InvitationId = "7ab938c5d52d7904ad09f9e7c20cc4eb"
)
## End(Not run)
```

---

**securitylake** *Amazon Security Lake*

**Description**

Amazon Security Lake is a fully managed security data lake service. You can use Security Lake to automatically centralize security data from cloud, on-premises, and custom sources into a data lake that’s stored in your Amazon Web Services account. Amazon Web Services Organizations is an account management service that lets you consolidate multiple Amazon Web Services accounts into an organization that you create and centrally manage. With Organizations, you can create member accounts and invite existing accounts to join your organization. Security Lake helps you
analyze security data for a more complete understanding of your security posture across the entire organization. It can also help you improve the protection of your workloads, applications, and data. The data lake is backed by Amazon Simple Storage Service (Amazon S3) buckets, and you retain ownership over your data.

Amazon Security Lake integrates with CloudTrail, a service that provides a record of actions taken by a user, role, or an Amazon Web Services service. In Security Lake, CloudTrail captures API calls for Security Lake as events. The calls captured include calls from the Security Lake console and code calls to the Security Lake API operations. If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon S3 bucket, including events for Security Lake. If you don’t configure a trail, you can still view the most recent events in the CloudTrail console in Event history. Using the information collected by CloudTrail you can determine the request that was made to Security Lake, the IP address from which the request was made, who made the request, when it was made, and additional details. To learn more about Security Lake information in CloudTrail, see the Amazon Security Lake User Guide.

Security Lake automates the collection of security-related log and event data from integrated Amazon Web Services and third-party services. It also helps you manage the lifecycle of data with customizable retention and replication settings. Security Lake converts ingested data into Apache Parquet format and a standard open-source schema called the Open Cybersecurity Schema Framework (OCSF).

Other Amazon Web Services and third-party services can subscribe to the data that’s stored in Security Lake for incident response and security data analytics.

Usage

```r
securitylake(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

• **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.

• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

### credentials
Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.

- **anonymous**: Set anonymous credentials.

### endpoint
Optional shorthand for complete URL to use for the constructed client.

### region
Optional shorthand for AWS Region used in instantiating the client.

### Value
A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

### Service syntax

```r
svc <- securitylake(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```
secret_access_key = "string",
session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

create_aws_log_source
create_custom_log_source
create_data_lake
create_data_lake_exception_subscription
create_data_lake_organization_configuration
create_subscriber
create_subscriber_notification
delete_aws_log_source
delete_custom_log_source
delete_data_lake
delete_data_lake_exception_subscription
delete_data_lake_organization_configuration
delete_subscriber
deregister_data_lake_delegated_administrator
get_data_lake_exception_subscription
get_data_lake_organization_configuration
get_data_lake_sources
get_subscriber
list_data_lake_exceptions
list_data_lakes
list_log_sources
list_subscribers
list_tags_for_resource
register_data_lake_delegated_administrator
tag_resource
untag_resource
update_data_lake
update_data_lake_exception_subscription
update_subscriber
update_subscriber_notification

Adds a natively supported Amazon Web Service as an Amazon Security Lake source
Adds a third-party custom source in Amazon Security Lake, from the Amazon Web Services Region where you want to create a custom source
Initializes an Amazon Security Lake instance with the provided (or default) configuration
Automatically enables Amazon Security Lake for new member accounts in your organization
Creates a subscription permission for accounts that are already enabled in Amazon Security Lake
Notifies the subscriber when new data is written to the data lake for the sources that the subscriber consumes
Removes a custom log source from Amazon Security Lake, to stop sending data from the custom source to Security Lake
When you disable Amazon Security Lake from your account, Security Lake is turned off
Deletes the specified notification subscription in Amazon Security Lake for the organization
Turns off automatic enablement of Amazon Security Lake for member accounts
Deletes the subscription permission and all notification settings for accounts that are already enabled in Amazon Security Lake
Deletes the specified notification subscription in Amazon Security Lake for the organization
Removes the custom log source from Amazon Security Lake, to stop sending data from the custom source to Security Lake
When you disable Amazon Security Lake from your account, Security Lake is turned off
Deletes the specified notification subscription in Amazon Security Lake for the organization
Retrieves the details of exception notifications for the account in Amazon Security Lake
Retrieves the configuration that will be automatically set up for accounts added to the organization
Retrieves a snapshot of the current Region, including whether Amazon Security Lake is enabled for those accounts
Retrieves the subscription information for the specified subscription ID
Lists the Amazon Security Lake exceptions that you can use to find the source of problems
Retrieves the Amazon Security Lake configuration object for the specified Amazon Web Services Region
Retrieves the log sources in the current Amazon Web Services Region
List all subscribers for the specific Amazon Security Lake account ID
Retrieves the tags (keys and values) that are associated with an Amazon Security Lake resource
Designates the Amazon Security Lake delegated administrator account for the organization
Adds or updates one or more tags that are associated with an Amazon Security Lake resource
Removes one or more tags (keys and values) from an Amazon Security Lake resource
Specifies where to store your security data and for how long
Updates the specified notification subscription in Amazon Security Lake for the organization
Updates an existing subscription for the given Amazon Security Lake account ID
Updates an existing notification method for the subscription (SQS or HTTPs endpoint)
## Examples

```r
## Not run:
svc <- securitylake()
svc$create_aws_log_source(
    Foo = 123
)
## End(Not run)
```

---

### serverlessapplicationrepository

**AWS\texttt{ServerlessApplicationRepository}**

---

### Description

The AWS Serverless Application Repository makes it easy for developers and enterprises to quickly find and deploy serverless applications in the AWS Cloud. For more information about serverless applications, see Serverless Computing and Applications on the AWS website.

The AWS Serverless Application Repository is deeply integrated with the AWS Lambda console, so that developers of all levels can get started with serverless computing without needing to learn anything new. You can use category keywords to browse for applications such as web and mobile backends, data processing applications, or chatbots. You can also search for applications by name, publisher, or event source. To use an application, you simply choose it, configure any required fields, and deploy it with a few clicks.

You can also easily publish applications, sharing them publicly with the community at large, or privately within your team or across your organization. To publish a serverless application (or app), you can use the AWS Management Console, AWS Command Line Interface (AWS CLI), or AWS SDKs to upload the code. Along with the code, you upload a simple manifest file, also known as the AWS Serverless Application Model (AWS SAM) template. For more information about AWS SAM, see AWS Serverless Application Model (AWS SAM) on the AWS Labs GitHub repository.

The AWS Serverless Application Repository Developer Guide contains more information about the two developer experiences available:

- **Consuming Applications** – Browse for applications and view information about them, including source code and readme files. Also install, configure, and deploy applications of your choosing.
  - Publishing Applications – Configure and upload applications to make them available to other developers, and publish new versions of applications.

### Usage

```r
serverlessapplicationrepository(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```
Arguments

config Optional configuration of credentials, endpoint, and/or region.

- credentials:
  - creds:
    - access_key_id: AWS access key ID
    - secret_access_key: AWS secret access key
    - session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.
- endpoint: The complete URL to use for the constructed client.
- region: The AWS Region used in instantiating the client.
- close_connection: Immediately close all HTTP connections.
- timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
- sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoint.html

credentials Optional credentials shorthand for the config parameter

- creds:
  - access_key_id: AWS access key ID
  - secret_access_key: AWS secret access key
  - session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

svc <- serverlessapplicationrepository(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string",
      ),
      profile = "string",
      anonymous = FALSE,
    ),
    endpoint = "string",
    region = "string",
    close_connection = FALSE,
    timeout = 60,
    s3_force_path_style = FALSE,
    sts_regional_endpoint = "string",
  ),
)
secret_access_key = "string",
session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string",
close_connection = "logical",
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

create_application
create_application_version
create_cloud_formation_change_set
create_cloud_formation_template
delete_application
get_application
get_application_policy
get_cloud_formation_template
list_application_dependencies
list_applications
list_application_versions
put_application_policy
unshare_application
update_application

Examples

## Not run:
svc <- serverlessapplicationrepository()
svc$create_application(
    Foo = 123
)

## End(Not run)

---

**servicecatalog**  
**AWS Service Catalog**

### Description

**Service Catalog**  
Service Catalog enables organizations to create and manage catalogs of IT services that are approved for Amazon Web Services. To get the most out of this documentation, you should be familiar with the terminology discussed in Service Catalog Concepts.

### Usage

```r
glue::glue confess
```

```r

```  

### Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:  
    - **creds**:  
      - **access_key_id**: AWS access key ID  
      - **secret_access_key**: AWS secret access key  
      - **session_token**: AWS temporary session token  
    - **profile**: The name of a profile to use. If not given, then the default profile is used.  
    - **anonymous**: Set anonymous credentials.  
  - **endpoint**: The complete URL to use for the constructed client.  
  - **region**: The AWS Region used in instantiating the client.  
  - **close_connection**: Immediately close all HTTP connections.  
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.  
  - **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.  
```
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy


**credentials**  Optional credentials shorthand for the config parameter

  • **creds**:
    – **access_key_id**: AWS access key ID
    – **secret_access_key**: AWS secret access key
    – **session_token**: AWS temporary session token
  
  • **profile**: The name of a profile to use. If not given, then the default profile is used.

  • **anonymous**: Set anonymous credentials.

**endpoint**  Optional shorthand for complete URL to use for the constructed client.

**region**  Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc(operation(...))`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- servicecatalog(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
```
anonymous = "logical"
),
  endpoint = "string",
  region = "string"
)

Operations

accept_portfolio_share
associate_budget_with_resource
associate_principal_with_portfolio
associate_product_with_portfolio
associate_service_action_with_provisioning_artifact
associate_tag_option_with_resource
batch_associate_service_action_with_provisioning_artifact
batch_disassociate_service_action_from_provisioning_artifact
copy_product
create_constraint
create_portfolio
create_portfolio_share
create_product
create_provisioned_product_plan
create_provisioning_artifact
create_service_action
create_tag_option
delete_constraint
delete_portfolio
delete_portfolio_share
delete_product
delete_provisioned_product_plan
delete_provisioning_artifact
delete_service_action
delete_tag_option
describe_constraint
describe_copy_product_status
describe_portfolio
describe_portfolio_shares
describe_portfolio_share_status
describe_product
describe_product_as_admin
describe_product_view
describe_provisioned_product
describe_provisioned_product_plan
describe_provisioning_artifact
describe_provisioning_parameters
describe_record
describe_service_action
describe_service_action_execution_parameters

Accepts an offer to share the specified portfolio
Associates the specified budget with the specified resource
Associates the specified principal ARN with the specified portfolio
Associates the specified product with the specified portfolio
Associates a self-service action with a provisioning artifact
Associate the specified TagOption with the specified portfolio
Associates multiple self-service actions with provisioning artifacts
Disassociates a batch of self-service actions from the specified provisioning artifact
Creates a constraint
Creates a portfolio
Shares the specified portfolio with the specified account or organization node
Creates a product
Creates a plan
Creates a provisioning artifact (also known as a version) for the specified product
Creates a self-service action
Creates a TagOption
Deletes the specified constraint
Deletes the specified portfolio
 Stops sharing the specified portfolio with the specified account or organization node
Deletes the specified product
Deletes the specified plan
Deletes the specified provisioning artifact (also known as a version) for the specified product
Deletes a self-service action
Deletes the specified TagOption
Gets information about the specified constraint
Gets the status of the specified copy product operation
Gets information about the specified portfolio
Returns a summary of each of the portfolio shares that were created for the specified portfolio
Gets the status of the specified portfolio share operation
Gets information about the specified product
Gets information about the specified product
Gets information about the specified product
Gets information about the specified provisioned product
Gets information about the resource changes for the specified provisioning artifact
Gets information about the specified provisioning artifact (also known as a version)
Gets information about the configuration required to provision a product
Describes a self-service action
Finds the default parameters for a specific self-service action
describe_tag_option
disable_aws_organizations_access
disassociate_budget_from_resource
disassociate_principal_from_portfolio
disassociate_product_from_portfolio
disassociate_service_action_from_provisioning_artifact
disassociate_tag_option_from_resource
enable_aws_organizations_access
execute_provisioned_product_plan
execute_provisioned_product_service_action
get_aws_organizations_access_status
get_provisioned_product_outputs
import_as_provisioned_product
list_accepted_portfolio_shares
list_budgets_for_resource
list_constraints_for_portfolio
list_launch_paths
list_organization_portfolio_access
list_portfolio_access
list_portfolios
list_portfolios_for_product
list_principals_for_portfolio
list_provisioned_product_plans
list_provisioning_artifacts
list_provisioning_artifacts_for_service_action
list_record_history
list_resources_for_tag_option
list_service_actions
list_service_actions_for_provisioning_artifact
list_stack_instances_for_provisioned_product
list_tag_options
notify_provision_product_engine_workflow_result
notify_terminate_provisioned_product_engine_workflow_result
notify_update_provisioned_product_engine_workflow_result
provision_product
reject_portfolio_share
scan_provisioned_products
search_products
search_products_as_admin
search_provisioned_products
terminate_provisioned_product
update_constraint
update_portfolio
update_portfolio_share
update_product
update_provisioned_product
update_provisioned_product_properties
update_provisioning_artifact

Gets information about the specified TagOption
Disable portfolio sharing through the Organizations service
Disassociates the specified budget from the specified resource
Disassociates a previously associated principal ARN from a portfolio
Disassociates the specified product from the specified portfolio
Disassociates the specified self-service action association from a portfolio
Disassociates the specified TagOption from the specified resource
Enable portfolio sharing feature through Organizations
Provisions or modifies a product based on the resource changes
Executes a self-service action against a provisioned product
Get the Access Status for Organizations portfolio share feature
This API takes either a ProvisionedProductId or a ProvisionedProductName
Requests the import of a resource as a Service Catalog provisioned product
Lists all imported portfolios for which account-to-account sharing has been accepted
Lists all the budgets associated to the specified resource
Lists the constraints for the specified portfolio and product
Lists the paths to the specified product
Lists the organization nodes that have access to the specified portfolio
Lists the account IDs that have access to the specified portfolio
Lists all portfolios in the catalog
Lists all portfolios that the specified product is associated with
Lists all PrincipalARNs and corresponding PrincipalTypes in the specified portfolio
Lists the plans for the specified provisioned product or all plans
Lists all provisioning artifacts (also known as versions) for the specified product
Lists all provisioning artifacts (also known as versions) for the specified portfolio
Lists the specified requests or all performed requests
Lists the resources associated with the specified TagOption
Lists all self-service actions
Returns a paginated list of self-service actions associated with the specified Product ID
Returns summary information about stack instances that are associated with the specified provisioned product
Lists the specified TagOptions or all TagOptions
Notifies the result of the provisioning engine execution
Notifies the result of the terminate engine execution
Notifies the result of the update engine execution
Provisions the specified product
Rejects an offer to share the specified portfolio
Lists the provisioned products that are available (not terminated)
Gets information about the products to which the caller has access
Gets information about the products for the specified portfolio
Gets information about the provisioned products that meet the specified criteria
Terminates the specified provisioned product
Updates the specified constraint
Updates the specified portfolio
Updates the specified portfolio share
Updates the specified product
Requests updates to the configuration of the specified provisioned product
Requests updates to the properties of the specified provisioned product
Updates the specified provisioning artifact (also known as a version)
update_service_action
update_tag_option

Examples

```r
## Not run:
svc <- servicecatalog()
svc$accept_portfolio_share(
  Foo = 123
)
## End(Not run)
```

servicediscovery

AWS Cloud Map

Description

Cloud Map

With Cloud Map, you can configure public DNS, private DNS, or HTTP namespaces that your microservice applications run in. When an instance becomes available, you can call the Cloud Map API to register the instance with Cloud Map. For public or private DNS namespaces, Cloud Map automatically creates DNS records and an optional health check. Clients that submit public or private DNS queries, or HTTP requests, for the service receive an answer that contains up to eight healthy records.

Usage

```r
servicediscovery(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- `config`: Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
    - `creds`:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
servicediscovery

– **profile**: The name of a profile to use. If not given, then the default profile is used.
– **anonymous**: Set anonymous credentials.

• **endpoint**: The complete URL to use for the constructed client.
• **region**: The AWS Region used in instantiating the client.
• **close_connection**: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

**credentials**
Optional credentials shorthand for the config parameter

• **creds**:
  – **access_key_id**: AWS access key ID
  – **secret_access_key**: AWS secret access key
  – **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.
• **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- servicediscovery(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
  ),
```
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

create_http_namespace Creates an HTTP namespace
create_private_dns_namespace Creates a private namespace based on DNS, which is visible only inside a specified Amazon VPC
create_public_dns_namespace Creates a public namespace based on DNS, which is visible on the internet
create_service Creates a service
delete_namespace Deletes a namespace from the current account
delete_service Deletes a specified service
deregister_instance Deletes the Amazon Route 53 DNS records and health check, if any, that Cloud Map created for the specified instance
discover_instances Discovers registered instances for a specified namespace and service
discover_instances_revision Discovers the increasing revision associated with an instance
get_instance Gets information about a specified instance
get_instances_health_status Gets the current health status (Healthy, Unhealthy, or Unknown) of one or more instances that are associated with a specified service
get_namespace Gets information about a namespace
get_operation Gets information about any operation that returns an operation ID in the response, such as a CreateHttpNamespace request
get_service Gets the settings for a specified service
list_instances Lists summary information about the instances that you registered by using a specified service
list_namespaces Lists summary information about the namespaces that were created by the current Amazon Web Services account
list_operations Lists operations that match the criteria that you specify
list_services Lists summary information for all the services that are associated with one or more namespaces
list_tags_for_resource Lists tags for the specified resource
register_instance Creates or updates one or more records and, optionally, creates a health check based on the settings in a specified service
untag_resource Removes one or more tags from the specified resource
update_http_namespace Updates an HTTP namespace
update_instance_custom_health_status Submits a request to change the health status of a custom health check to healthy or unhealthy
update_private_dns_namespace Updates a private DNS namespace
update_public_dns_namespace Updates a public DNS namespace
update_service Submits a request to perform the following operations:
Examples

```r
## Not run:
svc <- servicediscovery()
# This example creates an HTTP namespace.
svc$create_http_namespace(
  CreatorRequestId = "example-creator-request-id-0001",
  Description = "Example.com AWS Cloud Map HTTP Namespace",
  Name = "example-http.com"
)
```

## End(Not run)

---

### servicequotas

#### Service Quotas

**Description**

With Service Quotas, you can view and manage your quotas easily as your Amazon Web Services workloads grow. Quotas, also referred to as limits, are the maximum number of resources that you can create in your Amazon Web Services account. For more information, see the Service Quotas User Guide.

**Usage**

```r
servicequotas(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

**Arguments**

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      * **access_key_id**: AWS access key ID
      * **secret_access_key**: AWS secret access key
      * **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

• **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials**
Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.

- **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**
A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- servicequotas(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string",
  close_connection = "logical",
  timeout = "numeric",
  s3_force_path_style = "logical",
  sts_regional_endpoint = "string"
)
```

secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

associate_service_quota_template
delete_service_quota_increase_request_from_template
disassociate_service_quota_template
get_association_for_service_quota_template
get_aws_default_service_quota
get_requested_service_quota_change
get_service_quota
get_service_quota_increase_request_from_template
list_aws_default_service_quotas
list_requested_service_quota_change_history
list_requested_service_quota_change_history_by_quota
list_service_quota_increase_requests_in_template
list_service_quotas
list_services
list_tags_for_resource
put_service_quota_increase_request_into_template
request_service_quota_increase
tag_resource
untag_resource

Associates your quota request template with your organization
Deletes the quota increase request for the specified quota from your
Disables your quota request template
Retrieves the status of the association for the quota request template
Retrieves the default value for the specified quota
Retrieves information about the specified quota increase request
Retrieves the applied quota value for the specified quota
Retrieves information about the specified quota increase request in
Lists the default values for the quotas for the specified Amazon Web
Retrieves the quota increase requests for the specified Amazon Web
Retrieves the quota increase requests for the specified quota
Lists the quota increase requests in the specified quota request template
Lists the applied quota values for the specified Amazon Web Service
Lists the names and codes for the Amazon Web Services integrated
Returns a list of the tags assigned to the specified applied quota
Adds a quota increase request to your quota request template
Submits a quota increase request for the specified quota
Adds tags to the specified applied quota
Removes tags from the specified applied quota

Examples

```r
## Not run:
svc <- servicequotas()
svc$associate_service_quota_template(
    Foo = 123
)

## End(Not run)
```
Amazon Simple Email Service

Description

This document contains reference information for the Amazon Simple Email Service (Amazon SES) API, version 2010-12-01. This document is best used in conjunction with the Amazon SES Developer Guide.

For a list of Amazon SES endpoints to use in service requests, see Regions and Amazon SES in the Amazon SES Developer Guide.

This documentation contains reference information related to the following:

- Amazon SES API Actions
- Amazon SES API Data Types
- Common Parameters
- Common Errors

Usage

```
ses(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

- **credentials**: Optional credentials shorthand for the config parameter
• creds:
  – access_key_id: AWS access key ID
  – secret_access_key: AWS secret access key
  – session_token: AWS temporary session token
• profile: The name of a profile to use. If not given, then the default profile is used.
• anonymous: Set anonymous credentials.

endpoint  Optional shorthand for complete URL to use for the constructed client.
region    Optional shorthand for AWS Region used in instantiating the client.

Value
A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax
svc <- ses(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
Operations

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>clone_receipt_rule_set</td>
<td>Creates a receipt rule set by cloning an existing one</td>
</tr>
<tr>
<td>create_configuration_set</td>
<td>Creates a configuration set</td>
</tr>
<tr>
<td>create_configuration_set_event_destination</td>
<td>Creates a configuration set event destination</td>
</tr>
<tr>
<td>create_configuration_set_tracking_options</td>
<td>Creates an association between a configuration set and a custom domain</td>
</tr>
<tr>
<td>create_custom_verification_email_template</td>
<td>Creates a new custom verification email template</td>
</tr>
<tr>
<td>create_receipt_filter</td>
<td>Creates a new IP address filter</td>
</tr>
<tr>
<td>create_receipt_rule</td>
<td>Creates a receipt rule</td>
</tr>
<tr>
<td>create_receipt_rule_set</td>
<td>Creates an empty receipt rule set</td>
</tr>
<tr>
<td>create_template</td>
<td>Creates an email template</td>
</tr>
<tr>
<td>delete_configuration_set</td>
<td>Deletes a configuration set</td>
</tr>
<tr>
<td>delete_configuration_set_event_destination</td>
<td>Deletes a configuration set event destination</td>
</tr>
<tr>
<td>delete_configuration_set_tracking_options</td>
<td>Deletes an association between a configuration set and a custom domain</td>
</tr>
<tr>
<td>delete_custom_verification_email_template</td>
<td>Deletes an existing custom verification email template</td>
</tr>
<tr>
<td>delete_identity</td>
<td>Deletes the specified identity (an email address or a domain) from the list of verified identities.</td>
</tr>
<tr>
<td>delete_identity_policy</td>
<td>Deletes the specified sending authorization policy for the given identity</td>
</tr>
<tr>
<td>delete_receipt_filter</td>
<td>Deletes the specified IP address filter</td>
</tr>
<tr>
<td>delete_receipt_rule</td>
<td>Deletes the specified receipt rule</td>
</tr>
<tr>
<td>delete_receipt_rule_set</td>
<td>Deletes the specified receipt rule set and all of the receipt rules it contains.</td>
</tr>
<tr>
<td>delete_template</td>
<td>Deletes an email template</td>
</tr>
<tr>
<td>delete_verified_email_address</td>
<td>Deprecated</td>
</tr>
<tr>
<td>describe_active_receipt_rule_set</td>
<td>Returns the metadata and receipt rules for the receipt rule set that is currently active.</td>
</tr>
<tr>
<td>describe_configuration_set</td>
<td>Returns the details of the specified configuration set</td>
</tr>
<tr>
<td>describe_receipt_rule</td>
<td>Returns the details of the specified receipt rule</td>
</tr>
<tr>
<td>describe_receipt_rule_set</td>
<td>Returns the details of the specified receipt rule set</td>
</tr>
<tr>
<td>get_account_sending_enabled</td>
<td>Returns the email sending status of the Amazon SES account for the current region.</td>
</tr>
<tr>
<td>get_custom_verification_email_template</td>
<td>Returns the custom email verification template for the template name</td>
</tr>
<tr>
<td>get_identity_dkim_attributes</td>
<td>Returns the current status of Easy DKIM signing for an entity</td>
</tr>
<tr>
<td>get_identity_mail_from_domain_attributes</td>
<td>Returns the custom MAIL FROM attributes for a list of identities (email addresses or domains)</td>
</tr>
<tr>
<td>get_identity_notification_attributes</td>
<td>Returns the requested sending authorization policies for the given identity</td>
</tr>
<tr>
<td>get_identity_policies</td>
<td>Returns a list containing all of the identities (email addresses and/or domains), respectively.</td>
</tr>
<tr>
<td>get_identity_verification_attributes</td>
<td>Returns a list of sending authorization policies that are attached to the identity.</td>
</tr>
<tr>
<td>get_send_quota</td>
<td>Provides the sending limits for the Amazon SES account</td>
</tr>
<tr>
<td>get_send_statistics</td>
<td>Provides sending statistics for the current Amazon Web Services Region.</td>
</tr>
<tr>
<td>get_template</td>
<td>Displays the template object (which includes the Subject line, HTML, and text).</td>
</tr>
<tr>
<td>list_configuration_sets</td>
<td>Provides a list of the configuration sets associated with your Amazon SES account.</td>
</tr>
<tr>
<td>list_custom_verification_email_templates</td>
<td>Lists the existing custom verification email templates for your account.</td>
</tr>
<tr>
<td>list_identities</td>
<td>Returns a list containing all of the identities (email addresses and/or domains), respectively.</td>
</tr>
<tr>
<td>list_identity_policies</td>
<td>Returns a list of sending authorization policies that are attached to the identity.</td>
</tr>
<tr>
<td>list_receipt_filters</td>
<td>Lists the IP address filters associated with your Amazon Web Services account.</td>
</tr>
<tr>
<td>list_receipt_rule_sets</td>
<td>Lists the receipt rule sets that exist under your Amazon Web Services account.</td>
</tr>
<tr>
<td>list_templates</td>
<td>Lists the email templates present in your Amazon SES account in the current region.</td>
</tr>
<tr>
<td>list_verified_email_addresses</td>
<td>Deprecated</td>
</tr>
<tr>
<td>put_configuration_set_delivery_options</td>
<td>Adds or updates the delivery options for a configuration set</td>
</tr>
<tr>
<td>put_identity_policy</td>
<td>Adds or updates a sending authorization policy for the specified identity</td>
</tr>
<tr>
<td>reorder_receipt_rule_set</td>
<td>Reorders the receipt rules within a receipt rule set</td>
</tr>
<tr>
<td>send_bounce</td>
<td>Generates and sends a bounce message to the sender of an email you receive.</td>
</tr>
</tbody>
</table>
Composes an email message to multiple destinations

Adds an email address to the list of destinations for your Amazon SES account.

Composes an email message and immediately queues it for sending.

Composes an email message using an email template and immediately queues it for sending.

Sets the specified receipt rule set as the active receipt rule set.

Enables or disables Easy DKIM signing of email sent from an identity.

Given an identity (an email address or a domain), enables or disables whether Amazon SES forwards bounce and complaint notifications as email.

Given an identity (an email address or a domain), sets whether Amazon SES includes the original email headers in the Amazon Simple Notification Service (Amazon SNS) notifications of a specified type.

Enables or disables the custom MAIL FROM domain setup for a verified identity.

Sets an Amazon Simple Notification Service (Amazon SNS) topic to use when delivering notifications.

Sets the position of the specified receipt rule in the receipt rule set.

Creates a preview of the MIME content of an email when provided with a template and a set of replacement data.

Enables or disables email sending across your entire Amazon SES account.

Updates the event destination of a configuration set.

Enables or disables the publishing of reputation metrics for emails sent using a specific configuration set.

Enables or disables email sending for messages sent using a specific configuration set.

Modifies an association between a configuration set and a custom domain for open and click event tracking.

Updates an existing custom verification email template.

Updates a receipt rule.

Updates an email template.

Returns a set of DKIM tokens for a domain identity.

Deprecates a domain identity.

Adds an email address to the list of destinations for your Amazon SES account.

### Examples

```r
## Not run:
svc <- ses()

# The following example creates a receipt rule set by cloning an existing one:
svc$clone_receipt_rule_set(
  OriginalRuleSetName = "RuleSetToClone",
  RuleSetName = "RuleSetToCreate"
)

## End(Not run)
```

**Amazon Simple Email Service**

sesv2
Description

Amazon SES API v2

Amazon SES is an Amazon Web Services service that you can use to send email messages to your customers.

If you’re new to Amazon SES API v2, you might find it helpful to review the Amazon Simple Email Service Developer Guide. The Amazon SES Developer Guide provides information and code samples that demonstrate how to use Amazon SES API v2 features programmatically.

Usage

sesv2(config = list(), credentials = list(), endpoint = NULL, region = NULL)

Arguments

config

Optional configuration of credentials, endpoint, and/or region.

- credentials:
  - creds:
    - access_key_id: AWS access key ID
    - secret_access_key: AWS secret access key
    - session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.
- endpoint: The complete URL to use for the constructed client.
- region: The AWS Region used in instantiating the client.
- close_connection: Immediately close all HTTP connections.
- timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
- sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy

credentials

Optional credentials shorthand for the config parameter

- creds:
  - access_key_id: AWS access key ID
  - secret_access_key: AWS secret access key
  - session_token: AWS temporary session token
- profile: The name of a profile to use. If not given, then the default profile is used.
- anonymous: Set anonymous credentials.

endpoint

Optional shorthand for complete URL to use for the constructed client.

region

Optional shorthand for AWS Region used in instantiating the client.
A client for the service. You can call the service's operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
cvc <- sesv2(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

**Operations**

- `batch_get_metric_data`: Retrieves batches of metric data collected based on your sending activity
- `cancel_export_job`: Cancels an export job
- `create_configuration_set`: Create a configuration set
- `create_configuration_set_event_destination`: Create an event destination
- `create_contact`: Creates a contact, which is an end-user who is receiving the email, and adds them to a contact list
- `create_contact_list`: Creates a new custom verification email template
- `create_custom_verification_email_template`: Create a new pool of dedicated IP addresses
- `create_dedicated_ip_pool`
create_deliverability_test_report
create_email_identity
create_email_identity_policy
create_email_template
create_export_job
create_import_job
delete_configuration_set
delete_configuration_set_event_destination
delete_contact
delete_contact_list
delete_custom_verification_email_template
delete_dedicated_ip_pool
delete_email_identity
delete_email_identity_policy
delete_email_template
delete_suppressed_destination
get_account
get_blacklist_reports
get_configuration_set
get_configuration_set_event_destinations
get_contact
get_contact_list
get_custom_verification_email_template
get_dedicated_ip
get_dedicated_ip_pool
get_dedicated_ips
get_deliverability_dashboard_options
get_deliverability_test_report
get_domain_deliverability_campaign
get_domain_statistics_report
get_email_identity
get_email_identity_policies
get_email_template
get_export_job
get_import_job
get_message_insights
get_suppressed_destination
list_configuration_sets
list_contact_lists
list_contacts
list_custom_verification_email_templates
list_dedicated_ip_pools
list_deliverability_test_reports
list_domain_deliverability_campaigns
list_email_identities
list_email_templates
list_export_jobs
list_import_jobs
Create a new predictive inbox placement test
Starts the process of verifying an email identity
Creates the specified sending authorization policy for the given identity (an email address or a domain)
Creates an email template
Creates an export job for a data source and destination
Creates an import job for a data destination
Delete an existing configuration set
Delete an existing configuration set
Delete an event destination
Removes a contact from a contact list
Deletes a contact list and all of the contacts on that list
Deletes an existing custom verification email template
Delete a dedicated IP pool
Deletes an email identity
Deletes the specified sending authorization policy for the given identity (an email address or a domain)
Deletes an email template
Removes an email address from the suppression list for your account
Obtain information about the email-sending status and capabilities of your Amazon SES account
Retrieve a list of the blacklists that your dedicated IP addresses appear on
Get information about an existing configuration set, including the dedicated pool
Retrieve a list of event destinations that are associated with a configuration set
Returns a contact from a contact list
Returns contact list metadata
Returns the custom email verification template for the template name you specify
Get information about a dedicated IP address, including the name of the dedicated pool
List the dedicated IP addresses that are associated with your Amazon Web Services account
Retrieve information about the status of the Deliverability dashboard for your account
Retrieve the results of a predictive inbox placement test
Retrieve all the deliverability data for a specific campaign
Retrieve inbox placement and engagement rates for the domains that you use
Provides information about a specific identity, including the identity’s verification status, sending authorization policies, its DKIM authentication status, and its custom Mail-From settings
Returns the requested sending authorization policies for the given identity (an email address or a domain)
Displays the template object (which includes the subject line, HTML part and text part) for the template you specify
Provides information about an export job
Provides information about an import job
Provides information about a specific message, including the from address, subject, recipients, and any email tags associated with the message
Retrieves information about a specific email address that’s on the suppression list
List all of the configuration sets associated with your account in the current region
Lists all of the contacts available
Lists the contacts present in a specific contact list
Lists the existing custom verification email templates for your account in the current region
List all of the dedicated IP pools that exist in your Amazon Web Services account
Show a list of the predictive inbox placement tests that you’ve performed, regardless of their statuses
Lists all of the email identities that are associated with your Amazon SES account in the current region
Lists the email templates present in your Amazon SES account in the current region
Lists all of the export jobs
Lists all of the import jobs
list_recommendations
list_suppressed_destinations
list_tags_for_resource
put_account_dedicated_ip_warmup_attributes
put_account_details
put_account_sending_attributes
put_account_suppression_attributes
put_account_vdm_attributes
put_configuration_set_delivery_options
put_configuration_set_reputation_options
put_configuration_set_sending_options
put_configuration_set_tracking_options
put_configuration_set_vdm_options
put_dedicated_ip_in_pool
put_dedicated_ip_pool_scaling_attributes
put_dedicated_ip_warmup_attributes
put_deliverability_dashboard_option
put_email_identity_configuration_set_attributes
put_email_identity_dkim_attributes
put_email_identity_dkim_signing_attributes
put_email_identity_feedback_attributes
put_email_identity_mail_from_attributes
put_suppressed_destination
send_bulk_email
send_custom_verification_email
tag_resource
test_render_email_template
untag_resource
update_configuration_set_event_destination
update_contact
update_contact_list
update_custom_verification_email_template
update_email_identity_policy
update_email_template

Lists the recommendations present in your Amazon SES account in the current Amazon Web Services Region
Retrieves a list of email addresses that are on the suppression list for your account
Retrieve a list of the tags (keys and values) that are associated with a specified resource
Enable or disable the automatic warm-up feature for dedicated IP addresses
Update your Amazon SES account details
Enable or disable the ability of your account to send email
Change the settings for the account-level suppression list
Update your Amazon SES account VDM attributes
Associate a configuration set with a dedicated IP pool
Enable or disable collection of reputation metrics for emails that you send using a particular configuration set
Enable or disable email sending for messages that use a particular configuration set
Specify the account suppression list preferences for a configuration set
Specify a custom domain to use for open and click tracking elements in emails that use a particular configuration set
Specify VDM preferences for email that you send using the configuration set
Move a dedicated IP address to an existing dedicated IP pool
Used to convert a dedicated IP pool to a different scaling mode
Put dedicated ip warmup attributes
Enable or disable the Deliverability dashboard
Used to associate a configuration set with an email identity
Used to enable or disable DKIM authentication for an email identity
Used to configure or change the DKIM authentication settings for an email identity
Used to enable or disable feedback forwarding for an identity
Used to enable or disable the custom Mail-From domain configuration for an email identity
Adds an email address to the suppression list for your account
Composes an email message to multiple destinations
Adds an email address to the list of identities for your Amazon SES account
Sends an email message
Add one or more tags (keys and values) to a specified resource
Creates a preview of the MIME content of an email when provided with a template
Remove one or more tags (keys and values) from a specified resource
Update the configuration of an event destination for a configuration set
Updates a contact’s preferences for a list
Updates contact list metadata
Updates an existing custom verification email template
Updates the specified sending authorization email policy for the given identity (an email address or a domain)

Examples

```r
## Not run:
svc <- sesv2()

# Cancels the export job with ID ef28cf62-9d8e-4b60-9283-b09816c99a99
svc$cancel_export_job(
  JobId = "ef28cf62-9d8e-4b60-9283-b09816c99a99"
)

## End(Not run)
```
**Description**

Step Functions

Step Functions is a service that lets you coordinate the components of distributed applications and microservices using visual workflows.

You can use Step Functions to build applications from individual components, each of which performs a discrete function, or task, allowing you to scale and change applications quickly. Step Functions provides a console that helps visualize the components of your application as a series of steps. Step Functions automatically triggers and tracks each step, and retries steps when there are errors, so your application executes predictably and in the right order every time. Step Functions logs the state of each step, so you can quickly diagnose and debug any issues.

Step Functions manages operations and underlying infrastructure to ensure your application is available at any scale. You can run tasks on Amazon Web Services, your own servers, or any system that has access to Amazon Web Services. You can access and use Step Functions using the console, the Amazon Web Services SDKs, or an HTTP API. For more information about Step Functions, see the [Step Functions Developer Guide](#).

If you use the Step Functions API actions using Amazon Web Services SDK integrations, make sure the API actions are in camel case and parameter names are in Pascal case. For example, you could use Step Functions API action `startSyncExecution` and specify its parameter as `StateMachineArn`.

**Usage**

```r
sfn(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

**Arguments**

- `config` Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
    - `creds`:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
    - `anonymous`: Set anonymous credentials.
  - `endpoint`: The complete URL to use for the constructed client.
  - `region`: The AWS Region used in instantiating the client.
  - `close_connection`: Immediately close all HTTP connections.
**timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

**s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

**sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html

credentials Optional credentials shorthand for the config parameter

- **creds**:
  - `access_key_id`: AWS access key ID
  - `secret_access_key`: AWS secret access key
  - `session_token`: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.

- **anonymous**: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- sfn(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  )
)
```
secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

create_activity
create_state_machine
create_state_machine_alias
delete_activity
delete_state_machine
delete_state_machine_alias
delete_state_machine_version
describe_activity
describe_execution
describe_map_run
describe_state_machine
describe_state_machine_alias
describe_state_machine_for_execution
get_activity_task
get_execution_history
list_activities
list_executions
list_map_runs
list_state_machine_aliases
list_state_machines
list_state_machine_versions
list_tags_for_resource
publish_state_machine_version
redrive_execution
send_task_failure
send_task_heartbeat
send_task_success
start_execution
start_sync_execution
stop_execution
tag_resource
test_state
untag_resource
update_map_run
update_state_machine
update_state_machine_alias

Creates an activity
Creates a state machine
Creates an alias for a state machine that points to one or two versions of the same state machine
Deletes an activity
Deletes a state machine
Deletes a state machine alias
Deletes a state machine version
Describes an activity
Provides information about a state machine execution, such as the state machine associated with the execution, its execution role ARN, and relevant execution metadata
Provides information about a Map Run’s configuration, progress, and results
Provides information about a state machine’s definition, its IAM role Amazon Resource Name (ARN), and configuration
Returns details about a state machine alias
Provides information about a state machine’s definition, its execution role ARN, and configuration
Used by workers to retrieve a task (with the specified activity ARN) which has been scheduled for execution
Returns the history of the specified execution as a list of events
Lists the existing activities
Lists all executions of a state machine or a Map Run
Lists all Map Runs that were started by a given state machine execution
Lists aliases for a specified state machine ARN
Lists the existing state machines
Lists versions for the specified state machine Amazon Resource Name (ARN)
List tags for a given resource
Creates a version from the current revision of a state machine
Restarts unsuccessful executions of Standard workflows that didn’t complete successfully
Used by activity workers, Task states using the callback pattern, and optionally Task states using the job run pattern
Used by activity workers and Task states using the callback pattern, and optionally Task states using the job run pattern
Used by activity workers, Task states using the callback pattern, and optionally Task states using the job run pattern
Starts a state machine execution
Starts a Synchronous Express state machine execution
Stops an execution
Add a tag to a Step Functions resource
Accepts the definition of a single state and executes it
Remove a tag from a Step Functions resource
Updates an in-progress Map Run’s configuration to include changes to the settings that control maximum concurrency and Map Run failure
Updates an existing state machine by modifying its definition, roleArn, or loggingConfiguration
Updates the configuration of an existing state machine alias by modifying its description
validate_state_machine_definition \hspace{1em} Validates the syntax of a state machine definition

Examples

```r
## Not run:
svc <- sfn()
svc$create_activity(
  Foo = 123
)
## End(Not run)
```

Description

Shield Advanced

This is the Shield Advanced API Reference. This guide is for developers who need detailed information about the Shield Advanced API actions, data types, and errors. For detailed information about WAF and Shield Advanced features and an overview of how to use the WAF and Shield Advanced APIs, see the WAF and Shield Developer Guide.

Usage

```r
shield(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• `s3_force_path_style`: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.

• `sts_regional_endpoint`: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials** Optional credentials shorthand for the config parameter

• `creds`:
  – `access_key_id`: AWS access key ID
  – `secret_access_key`: AWS secret access key
  – `session_token`: AWS temporary session token

• `profile`: The name of a profile to use. If not given, then the default profile is used.

• `anonymous`: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- shield(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    )
  )
)
```
Operations

- **associate_drt_log_bucket**
  Authorizes the Shield Response Team (SRT) to access the specified Amazon S3 bucket containing log data such as Application Load Balancer access logs, CloudFront logs, or logs from third party sources

- **associate_drt_role**
  Authorizes the Shield Response Team (SRT) using the specified role, to access your Amazon Web Services account to assist with DDoS attack mitigation during potential attacks

- **associate_health_check**
  Adds health-based detection to the Shield Advanced protection for a resource

- **associate_proactive_engagement_details**
  Initializes proactive engagement and sets the list of contacts for the Shield Response Team (SRT) to use

- **create_protection**
  Enables Shield Advanced for a specific Amazon Web Services resource

- **create_protection_group**
  Creates a grouping of protected resources so they can be handled as a collective

- **create_subscription**
  Activates Shield Advanced for an account

- **delete_protection**
  Deletes an Shield Advanced Protection

- **delete_protection_group**
  Removes the specified protection group

- **delete_subscription**
  Removes Shield Advanced from an account

- **describe_attack**
  Describes the details of a DDoS attack

- **describe_attack_statistics**
  Provides information about the number and type of attacks Shield has detected in the last year for all resources that belong to your account, regardless of whether you've defined Shield protections for them

- **describe_drt_access**
  Returns the current role and list of Amazon S3 log buckets used by the Shield Response Team (SRT) to access your Amazon Web Services account while assisting with attack mitigation

- **describe_emergency_contact_settings**
  A list of email addresses and phone numbers that the Shield Response Team (SRT) can use to contact you if you have proactive engagement enabled, for escalations to the SRT and to initiate proactive customer support

- **describe_protection**
  Lists the details of a Protection object

- **describe_protection_group**
  Returns the specification for the specified protection group

- **describe_subscription**
  Provides details about the Shield Advanced subscription for an account

- **disable_application_layer_automatic_response**
  Disable the Shield Advanced automatic application layer DDoS mitigation feature for the protected resource

- **disable_proactive_engagement**
  Removes authorization from the Shield Response Team (SRT) to notify contacts about escalations to the SRT and to initiate proactive customer support

- **disassociate_drt_log_bucket**
  Removes the Shield Response Team’s (SRT) access to the specified Amazon S3 log bucket that you shared previously

- **disassociate_drt_role**
  Removes the Shield Response Team’s (SRT) access to your Amazon Web Services account

- **disassociate_health_check**
  Removes health-based detection from the Shield Advanced protection for a resource

- **enable_application_layer_automatic_response**
  Enable the Shield Advanced automatic application layer DDoS mitigation for the protected resource

- **enable_proactive_engagement**
  Authorizes the Shield Response Team (SRT) to use email and phone to notify contacts about escalations to the SRT and to initiate proactive customer support

- **get_subscription_state**
  Returns the SubscriptionState, either Active or Inactive

- **list_attacks**
  Returns all ongoing DDoS attacks or all DDoS attacks during a specified time period

- **list_protection_groups**
  Retrieves ProtectionGroup objects for the account

- **list_protections**
  Retrieves Protection objects for the account

- **list_resources_in_protection_group**
  Retrieves the resources that are included in the protection group

- **list_tags_for_resource**
  Retrieves tags from a resource in Shield

- **tag_resource**
  Adds or updates tags for a resource in Shield

- **untag_resource**
  Removes tags from a resource in Shield

- **update_application_layer_automatic_response**
  Updates an existing Shield Advanced automatic application layer DDoS mitigation configuration for the specified resource

- **update_emergency_contact_settings**
  Updates the details of the list of email addresses and phone numbers that the Shield Response Team (SRT) can use to contact you if you have proactive engagement enabled, for escalations to the SRT and to initiate proactive customer support

- **update_protection_group**
  Updates an existing protection group

- **update_subscription**
  Updates the details of an existing subscription
simpledb

Amazon SimpleDB

Description

Amazon SimpleDB is a web service providing the core database functions of data indexing and querying in the cloud. By offloading the time and effort associated with building and operating a web-scale database, SimpleDB provides developers the freedom to focus on application development.

A traditional, clustered relational database requires a sizable upfront capital outlay, is complex to design, and often requires extensive and repetitive database administration. Amazon SimpleDB is dramatically simpler, requiring no schema, automatically indexing your data and providing a simple API for storage and access. This approach eliminates the administrative burden of data modeling, index maintenance, and performance tuning. Developers gain access to this functionality within Amazon’s proven computing environment, are able to scale instantly, and pay only for what they use.


Usage

```
simpledb(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:  
    - **creds**:  
      - **access_key_id**: AWS access key ID  
      - **secret_access_key**: AWS secret access key  
      - **session_token**: AWS temporary session token  
    - **profile**: The name of a profile to use. If not given, then the default profile is used.  
    - **anonymous**: Set anonymous credentials.  
  - **endpoint**: The complete URL to use for the constructed client.  
  - **region**: The AWS Region used in instantiating the client.

Examples

```r
## Not run:
svc <- shield()
svc$associate_drt_log_bucket(
  Foo = 123
)
## End(Not run)
```
• **close_connection**: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

**credentials**  Optional credentials shorthand for the config parameter

• **creds**:
  – **access_key_id**: AWS access key ID
  – **secret_access_key**: AWS secret access key
  – **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.
• **anonymous**: Set anonymous credentials.

**endpoint**  Optional shorthand for complete URL to use for the constructed client.

**region**  Optional shorthand for AWS Region used in instantiating the client.

**Value**
A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- simpledb(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
```
creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

batch_delete_attributes Performs multiple DeleteAttributes operations in a single call, which reduces round trips and latencies
batch_put_attributes The BatchPutAttributes operation creates or replaces attributes within one or more items
create_domain The CreateDomain operation creates a new domain
delete_attributes Deletes one or more attributes associated with an item
delete_domain The DeleteDomain operation deletes a domain
domain_metadata Returns information about the domain, including when the domain was created, the number of items
get_attributes Returns all of the attributes associated with the specified item
list_domains The ListDomains operation lists all domains associated with the Access Key ID
put_attributes The PutAttributes operation creates or replaces attributes in an item
select The Select operation returns a set of attributes for ItemNames that match the select expression

Examples

```r
## Not run:
svc <- simpledb()
svc$batch_delete_attributes(
    Foo = 123
)
## End(Not run)
```

**sns**

Amazon Simple Notification Service

Description

Amazon Simple Notification Service (Amazon SNS) is a web service that enables you to build distributed web-enabled applications. Applications can use Amazon SNS to easily push real-time notification messages to interested subscribers over multiple delivery protocols. For more information about this product see the Amazon SNS product page. For detailed information about Amazon SNS features and their associated API calls, see the Amazon SNS Developer Guide.
For information on the permissions you need to use this API, see Identity and access management in Amazon SNS in the Amazon SNS Developer Guide.

We also provide SDKs that enable you to access Amazon SNS from your preferred programming language. The SDKs contain functionality that automatically takes care of tasks such as: cryptographically signing your service requests, retrying requests, and handling error responses. For a list of available SDKs, go to Tools for Amazon Web Services.

Usage

```r
sns(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

- `config`: Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
    - `creds`:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
    - `anonymous`: Set anonymous credentials.
  - `endpoint`: The complete URL to use for the constructed client.
  - `region`: The AWS Region used in instantiating the client.
  - `close_connection`: Immediately close all HTTP connections.
  - `timeout`: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - `s3_force_path_style`: Set this to true to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
  - `sts_regional_endpoint`: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html

- `credentials`: Optional credentials shorthand for the config parameter
  - `creds`:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - `profile`: The name of a profile to use. If not given, then the default profile is used.
  - `anonymous`: Set anonymous credentials.

- `endpoint`: Optional shorthand for complete URL to use for the constructed client.

- `region`: Optional shorthand for AWS Region used in instantiating the client.
Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- sns(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `add_permission`: Adds a statement to a topic’s access control policy, granting access for the specified Amazon Web Services accounts.
- `check_if_phone_number_is_opted_out`: Accepts a phone number and indicates whether the phone holder has opted out of receiving SMS messages.
- `confirm_subscription`: Verifies an endpoint owner’s intent to receive messages by validating the token sent in an earlier Subscribe action.
- `create_platform_application`: Creates a platform application object for one of the supported push notification services, such as APNS and GCM (Firebase Cloud Messaging), to which devices and mobile apps may register.
- `create_platform_endpoint`: Creates an endpoint for a device and mobile app on one of the supported push notification services, such as GCM (Firebase Cloud Messaging) and APNS.
- `create_sms_sandbox_phone_number`: Adds a destination phone number to an Amazon Web Services account in the SMS sandbox and sends a one-time password (OTP) to that phone number.
- `create_topic`: Creates a topic to which notifications can be published.
- `delete_endpoint`: Deletes the endpoint for a device and mobile app from Amazon SNS.
delete_platform_application
delete_sms_sandbox_phone_number
delete_topic
get_data_protection_policy
get_endpoint_attributes
get_platform_application_attributes
get_sms_attributes
get_sms_sandbox_account_status
get_subscription_attributes
get_topic_attributes
list_endpoints_by_platform_application
list_origination_numbers
list_phone_numbers_opted_out
list_platform_applications
list_sms_sandbox_phone_numbers
list_subscriptions
list_subscriptions_by_topic
list_tags_for_resource
list_topics
opt_in_phone_number
publish
publish_batch
put_data_protection_policy
remove_permission
set_endpoint_attributes
set_platform_application_attributes
set_sms_attributes
set_subscription_attributes
set_topic_attributes
subscribe
tag_resource
unsubscribe
untag_resource
verify_sms_sandbox_phone_number

Deletes a platform application object for one of the supported push notification services.
Deletes an Amazon Web Services account’s verified or pending phone number from the SMS sandbox.
Deletes a topic and all its subscriptions.
Retrieves the specified inline DataProtectionPolicy document that is stored in the specified Amazon SNS topic.
Retrieves the endpoint attributes for a device on one of the supported push notification services.
Retrieves the attributes of the platform application object for the supported push notification services.
Returns the settings for sending SMS messages from your Amazon Web Services account.
Retrieves the SMS sandbox status for the calling Amazon Web Services account in the target Amazon Web Services Region.
Returns all of the properties of a subscription.
Returns all of the properties of a topic.
Lists the endpoints and endpoint attributes for devices in a supported push notification service.
Lists the calling Amazon Web Services account’s dedicated origination numbers and their metadata.
Returns a list of phone numbers that are opted out, meaning you cannot send SMS messages to them.
Lists the platform application objects for the supported push notification services, such as APNS and GCM.
Lists the calling Amazon Web Services account’s current verified and pending destination phone numbers in the SMS sandbox.
Returns a list of the requester’s subscriptions.
Returns a list of the subscriptions to a specific topic.
List all tags added to the specified Amazon SNS topic.
Returns a list of the requester’s topics.
Use this request to opt in a phone number that is opted out, which enables you to resume sending SMS messages to the number.
Sends a message to an Amazon SNS topic, a text message (SMS message) directly to a phone number, or a message to a mobile platform endpoint (when you specify the TargetArn).
Publishes up to ten messages to the specified topic.
Adds or updates an inline policy document that is stored in the specified Amazon SNS topic.
Removes a statement from a topic’s access control policy.
Sets the attributes for an endpoint for a device on one of the supported push notification services.
Sets the attributes of the platform application object for the supported push notification services.
Use this request to set the default settings for sending SMS messages and receiving SMS usage reports from Amazon Web Services.
Allow a subscription owner to set an attribute of the subscription to a new value.
Allows a topic owner to set an attribute of the topic to a new value.
Subscribes an endpoint to an Amazon SNS topic.
Add tags to the specified Amazon SNS topic.
Deletes a subscription.
Remove tags from the specified Amazon SNS topic.
Verifies a destination phone number with a one-time password (OTP) for the calling Amazon Web Services account.

Examples

```r
## Not run:
svc <- sns()
svc$add_permission(
    Foo = 123
)

## End(Not run)
```
Amazon Simple Queue Service

Description

Welcome to the Amazon SQS API Reference.

Amazon SQS is a reliable, highly-scalable hosted queue for storing messages as they travel between applications or microservices. Amazon SQS moves data between distributed application components and helps you decouple these components.

For information on the permissions you need to use this API, see Identity and access management in the Amazon SQS Developer Guide.

You can use Amazon Web Services SDKs to access Amazon SQS using your favorite programming language. The SDKs perform tasks such as the following automatically:

- Cryptographically sign your service requests
- Retry requests
- Handle error responses

Additional information

- Amazon SQS Product Page
- Amazon SQS Developer Guide
  - Making API Requests
  - Amazon SQS Message Attributes
  - Amazon SQS Dead-Letter Queues
- Amazon SQS in the Command Line Interface
- Amazon Web Services General Reference
  - Regions and Endpoints

Usage

sqs(config = list(), credentials = list(), endpoint = NULL, region = NULL)

Arguments

config Optional configuration of credentials, endpoint, and/or region.

- credentials:
  - creds:
    - access_key_id: AWS access key ID
    - secret_access_key: AWS secret access key
    - session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
– **anonymous**: Set anonymous credentials.

- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials**
Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the `Operations` section.

**Service syntax**

```r
svc <- sqs(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
  ),
```


sts_regional_endpoint = "string"
),
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

add_permission Adds a permission to a queue for a specific principal
cancel_message_move_task Cancels a specified message movement task
change_message_visibility Changes the visibility timeout of a specified message in a queue to a new value
change_message_visibility_batch Changes the visibility timeout of multiple messages
create_queue Creates a new standard or FIFO queue
delete_message Deletes the specified message from the specified queue
delete_message_batch Deletes up to ten messages from the specified queue
delete_queue Deletes the queue specified by the QueueUrl, regardless of the queue’s contents
get_queue_attributes Gets attributes for the specified queue
get_queue_url Returns the URL of an existing Amazon SQS queue
list_dead_letter_source_queues Returns a list of your queues that have the RedrivePolicy queue attribute configured with a dead-letter queue
list_message_move_tasks Gets the most recent message movement tasks (up to 10) under a specific source queue
list_queues Returns a list of your queues in the current region
list_queue_tags List all cost allocation tags added to the specified Amazon SQS queue
purge_queue Deletes available messages in a queue (including in-flight messages) specified by the QueueUrl parameter
receive_message Retrieves one or more messages (up to 10), from the specified queue
remove_permission Revokes any permissions in the queue policy that matches the specified Label parameter
send_message Delivers a message to the specified queue
send_message_batch You can use SendMessageBatch to send up to 10 messages to the specified queue by assigning either identical or different values to each message (or by not assigning values at all)
set_queue_attributes Sets the value of one or more queue attributes
start_message_move_task Starts an asynchronous task to move messages from a specified source queue to a specified destination queue
tag_queue Add cost allocation tags to the specified Amazon SQS queue
untag_queue Remove cost allocation tags from the specified Amazon SQS queue

Examples

## Not run:
svc <- sqs()
svc$add_permission(
Amazon Simple Systems Manager (SSM)

Description

Amazon Web Services Systems Manager is the operations hub for your Amazon Web Services applications and resources and a secure end-to-end management solution for hybrid cloud environments that enables safe and secure operations at scale.

This reference is intended to be used with the Amazon Web Services Systems Manager User Guide. To get started, see Setting up Amazon Web Services Systems Manager.

Related resources

- For information about each of the capabilities that comprise Systems Manager, see Systems Manager capabilities in the Amazon Web Services Systems Manager User Guide.
- For details about predefined runbooks for Automation, a capability of Amazon Web Services Systems Manager, see the Systems Manager Automation runbook reference.
- For information about AppConfig, a capability of Systems Manager, see the AppConfig User Guide and the * AppConfig API Reference*.
- For information about Incident Manager, a capability of Systems Manager, see the Systems Manager Incident Manager User Guide and the * Systems Manager Incident Manager API Reference*.

Usage

```r
ssm(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
• **close_connection**: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

**credentials** Optional credentials shorthand for the config parameter

  • **creds:**
    – **access_key_id**: AWS access key ID
    – **secret_access_key**: AWS secret access key
    – **session_token**: AWS temporary session token
  • **profile**: The name of a profile to use. If not given, then the default profile is used.
  • **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
csvc <- ssm(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
```
```python
creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

add_tags_to_resource
associate_ops_item_related_item
cancel_command
cancel_maintenance_window_execution
create_activation
create_association
create_association_batch
create_document
create_maintenance_window
create_ops_item
create_ops_metadata
create_patch_baseline
create_resource_data_sync
delete_activation
delete_association
delete_document
delete_inventory
delete_maintenance_window
delete_ops_item
delete_ops_metadata
delete_parameter
delete_parameters
delete_patch_baseline
delete_resource_data_sync
delete_resource_policy
deregisterManagedInstance
deregister_patch_baseline_for_patch_group
deregister_target_from_maintenance_window
deregister_task_from_maintenance_window
describe_activations
describe_association
describe_association_executions
describe_association_execution_targets
describe_automation_executions
```

Adds or overwrites one or more tags for the specified resource.
Associates a related item to a Systems Manager OpsCenter OpsItem.
Attempts to cancel the command specified by the Command ID.
Stops a maintenance window execution that is already in progress.
Generates an activation code and activation ID you can use to register your on-premises server, edge device, or virtual machine with Amazon Web Services Systems Manager.
Associates the specified Amazon Web Services Systems Manager (SSM) patch baseline with the specified managed nodes.
Creates a new association.
Creates a new maintenance window.
Creates a new OpsItem.
If you create a new application in Application Manager, Amazon Web Services Systems Manager calls this API operation to specify information about the new application.
A Patch Manager association defines the state that you want to maintain on your managed nodes.
Deletes an activation.
Disassociates the specified Amazon Web Services Systems Manager document from the specified managed node.
Deletes the Amazon Web Services Systems Manager document.
Delete a custom inventory type or the data associated with a custom Inventory type.
Deletes a maintenance window.
Delete an OpsItem.
Delete OpsMetadata related to an application.
Delete a parameter from the system.
Delete a list of parameters.
Deletes a patch baseline.
Deletes a resource data sync configuration.
Deletes a Systems Manager resource policy.
Removes the server or virtual machine from the list of registered systems.
Removes a patch group from a patch baseline.
Removes a target from a maintenance window.
Removes a task from a maintenance window.
Describes details about the activation, such as the date and time the activation was created.
Describes the association for the specified target or managed node.
Views all executions for a specific association ID.
Views information about a specific execution of a specific association ID.
describe_automation_step_executions
describe_available_patches
describe_document
describe_document_permission
describe_effective_instance_associations
describe_effective_patches_for_patch_baseline
describe_instance_associations_status
describe_instance_information
describe_instance_patches
describe_instance_patch_states
describe_instance_patch_states_for_patch_group
describe_instance_properties
describe_inventory_deletions
describe_maintenance_window_executions
describe_maintenance_window_execution_task_invocations
describe_maintenance_window_execution_tasks
describe_maintenance_windows
describe_maintenance_window_schedule
describe_maintenance_windows_for_target
describe_maintenance_window_targets
describe_maintenance_window_tasks
describe_parameters
describe_patch_baselines
describe_patch_groups
describe_patch_group_state
describe_patch_properties
describe_sessions
disassociate_ops_item_related_item
get_automation_execution
get_calendar_state
get_command_invocation
get_connection_status
get_default_patch_baseline
get_deployable_patch_snapshot_for_instance
get_document
get_inventory
get_inventory_schema
get_maintenance_window
get_maintenance_window_execution
get_maintenance_window_execution_task
get_maintenance_window_execution_task_invocation
get_maintenance_window_task
get_ops_item
get_ops_metadata
get_ops_summary
get_parameter
get_parameter_history

Information about all active and terminated step executions in a workflow.
Lists all patches eligible to be included in a patch baseline.
Describes the specified Amazon Web Services Systems Manager (SSM) document.
Describes the permissions for a Amazon Web Services Systems Manager (SSM) document.
All associations for the managed nodes.
Retrieves the current effective patches (the patch and the approval state) for the managed nodes.
The status of the associations for the managed nodes.
Provides information about one or more of your managed nodes.
Retrieves information about the patches on the specified managed node.
Retrieves the high-level patch state of one or more managed nodes.
Retrieves the high-level patch state for the managed nodes in the patch baseline.
An API operation used by the Systems Manager console to display information about your managed nodes.
Describes a specific delete inventory operation.
Lists the executions of a maintenance window.
Retrieves the individual task executions (one per target) for a particular maintenance window.
For a given maintenance window execution, lists the tasks that ran as part of the execution.
Retrieves the maintenance windows in an Amazon Web Services account.
Retrieves information about upcoming executions of a maintenance window.
Retrieves information about the maintenance window targets or tasks.
Lists the targets registered with the maintenance window.
Lists the tasks in a maintenance window.
Query a set of OpsItems.
Lists the parameters in your Amazon Web Services account or associated with a particular set of permissions.
Lists the patch baselines in your Amazon Web Services account.
Lists all patch groups that have been registered with patch baselines.
Returns high-level aggregated patch compliance state information for a patch group.
Lists the properties of available patches organized by product, product family, classification, severity, and other properties of available patches.
Retrieves a list of all active sessions (both connected and disconnected).
Deletes the association between an OpsItem and a related item.
Get detailed information about a particular Automation execution.
Gets the state of an Automation execution.
Returns detailed information about command execution for an invocation.
Retrieves the Session Manager connection status for a managed node.
Retrieves the default patch baseline.
Retrieves the current snapshot for the patch baseline managed node.
Retrieves the contents of the specified Amazon Web Services Systems Manager document.
Return a list of inventory type names for the account, or return an object containing the name, description, and other attributes of the inventory type.
Retrieves a maintenance window.
Retrieves details about a specific maintenance window execution.
Retrieves the details about a specific task run as part of a maintenance window execution.
Retrieves information about a specific task running on a specified managed node.
Retrieves the details of a maintenance window task.
Get information about an OpsItem by using the ID.
View operational metadata related to an application in Application Manager.
View a summary of operations metadata (OpsData) based on specified filters.
Get information about a single parameter by specifying the parameter name.
Retrieves the history of all changes to a parameter.
get_parameters
get_parameters_by_path
get_patch_baseline
get_patch_baseline_for_patch_group
get_resource_policies
get_service_setting
label_parameter_version
list_associations
list_association_versions
list_command_invocations
list_commands
list_compliance_items
list_compliance_summaries
list_document_metadata_history
list_documents
list_document_versions
list_inventory_entries
list_ops_item_events
list_ops_item_related_items
list_ops_metadata
list_resource_compliance_summaries
list_resource_data_sync
list_tags_for_resource
modify_document_permission
put_compliance_items
put_inventory
put_parameter
put_resource_policy
register_default_patch_baseline
register_patch_baseline_for_patch_group
register_target_with_maintenance_window
register_task_with_maintenance_window
remove_tags_from_resource
reset_service_setting
resume_session
send_automation_signal
send_command
start_associations_once
start_automation_execution
start_change_request_execution
start_session
stop_automation_execution
terminate_session
unlabel_parameter_version
update_association
update_association_status
update_document
update_document_default_version

Get information about one or more parameters by specifying multiple parameter names.
Retrieve information about one or more parameters in a specific hierarchy.
Retrieves information about a patch baseline.
Retrieves the patch baseline that should be used for the specified patch group.
Returns an array of the Policy object.
ServiceSetting is an account-level setting for an Amazon Web Services service.
A parameter label is a user-defined alias to help you manage different versions of a parameter.
Returns all State Manager associations in the current Amazon Web Services account and Region.
Retrieves all versions of an association for a specific association ID.
An invocation is copy of a command sent to a specific managed node.
Lists the commands requested by users of the Amazon Web Services account.
For a specified resource ID, this API operation returns a list of compliance types.
Returns a summary count of compliant and non-compliant resources.
Information about approval reviews for a version of a change template in Change Manager.
Returns all Systems Manager (SSM) documents in the current Amazon Web Services account and Region.
List all versions for a document.
A list of inventory items returned by the request.
Returns a list of all OpsItem events in the current Amazon Web Services Region.
Lists all related-item resources associated with a Systems Manager association.
Amazon Web Services Systems Manager calls this API operation when displaying all Application Manager OpsMetadata objects or blobs.
Returns a resource-level summary count.
Lists your resource data sync configurations.
Returns a list of the tags assigned to the specified resource.
Shares a Amazon Web Services Systems Manager document (SSM document) publicly or privately.
Registers a compliance type and other compliance details on a designated resource.
Bulk update custom inventory items on one or more managed nodes.
Add a parameter to the system.
Creates or updates a Systems Manager resource policy.
Defines the default patch baseline for the relevant operating system.
Registers a patch baseline for a patch group.
Registers a target with a maintenance window.
Adds a new task to a maintenance window.
Removes tag keys from the specified resource.
ServiceSetting is an account-level setting for an Amazon Web Services service.
Reconnects a session to a managed node after it has been disconnected.
Sends a signal to an Automation execution to change the current behavior or status of the execution.
Runs commands on one or more managed nodes.
Runs an association immediately and only once.
Initiates execution of an Automation runbook.
Creates a change request for Change Manager.
Initiates a connection to a target (for example, a managed node).
Stop an Automation that is currently running.
Permanently ends a session and closes the data connection between the SSM Agent on a managed node.
Remove a label or labels from a parameter.
Updates an association.
Updates the status of the Amazon Web Services Systems Manager association.
Updates one or more values for an SSM document.
Set the default version of a document.
update_document_metadata
update_maintenance_window
update_maintenance_window_target
update_maintenance_window_task
update_managed_instance_role
update_ops_item
update_ops_metadata
update_patch_baseline
update_resource_data_sync
update_service_setting

Examples

```r
## Not run:
svc <- ssm()
svc$add_tags_to_resource(
  Foo = 123
)

## End(Not run)
```

**ssmcontacts**

**AWS Systems Manager Incident Manager Contacts**

**Description**

Systems Manager Incident Manager is an incident management console designed to help users mitigate and recover from incidents affecting their Amazon Web Services-hosted applications. An incident is any unplanned interruption or reduction in quality of services.

Incident Manager increases incident resolution by notifying responders of impact, highlighting relevant troubleshooting data, and providing collaboration tools to get services back up and running. To achieve the primary goal of reducing the time-to-resolution of critical incidents, Incident Manager automates response plans and enables responder team escalation.

**Usage**

```r
ssmcontacts(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```
ssmcontacts

Arguments

config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

credentials
  endpoint Optional shorthand for complete URL to use for the constructed client.
  region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

svc <- ssmcontacts(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string",
      ),
      profile = "string",
      anonymous = FALSE,
    ),
    endpoint = "string",
    region = "string",
    close_connection = FALSE,
    timeout = 60,
    s3_force_path_style = FALSE,
    sts_regional_endpoint = "string",
  ),
)
```python
secret_access_key = "string",
session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string",
close_connection = "logical",
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
  creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
  ),
  profile = "string",
anonymous = "logical"
  ),
endpoint = "string",
region = "string"
)
```

## Operations

<table>
<thead>
<tr>
<th>Operation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>accept_page</td>
<td>Used to acknowledge an engagement to a contact channel during an incident</td>
</tr>
<tr>
<td>activate_contact_channel</td>
<td>Activates a contact's contact channel</td>
</tr>
<tr>
<td>create_contact</td>
<td>Contacts are either the contacts that Incident Manager engages during an incident or the escalation plans that Incident Manager uses to engage contacts in phases during an incident</td>
</tr>
<tr>
<td>create_contact_channel</td>
<td>A contact channel is the method that Incident Manager uses to engage your contact</td>
</tr>
<tr>
<td>create_rotation</td>
<td>Creates a rotation in an on-call schedule</td>
</tr>
<tr>
<td>create_rotation_override</td>
<td>Creates an override for a rotation in an on-call schedule</td>
</tr>
<tr>
<td>deactivate_contact_channel</td>
<td>To no longer receive Incident Manager engagements to a contact channel, you can deactivate the channel</td>
</tr>
<tr>
<td>delete_contact</td>
<td>To remove a contact from Incident Manager, you can delete the contact</td>
</tr>
<tr>
<td>delete_contact_channel</td>
<td>To no longer receive engagements on a contact channel, you can delete the channel from a contact</td>
</tr>
<tr>
<td>delete_rotation</td>
<td>Deletes a rotation from the system</td>
</tr>
<tr>
<td>delete_rotation_override</td>
<td>Deletes an existing override for an on-call rotation</td>
</tr>
<tr>
<td>describe_engagement</td>
<td>Incident Manager uses engagements to engage contacts and escalation plans during an incident</td>
</tr>
<tr>
<td>describe_page</td>
<td>Lists details of the engagement to a contact channel</td>
</tr>
<tr>
<td>get_contact</td>
<td>Retrieves information about the specified contact or escalation plan</td>
</tr>
<tr>
<td>get_contact_channel</td>
<td>List details about a specific contact channel</td>
</tr>
<tr>
<td>get_contact_policy</td>
<td>Retrieves the resource policies attached to the specified contact or escalation plan</td>
</tr>
<tr>
<td>get_rotation</td>
<td>Retrieves information about an on-call rotation</td>
</tr>
<tr>
<td>get_rotation_override</td>
<td>Retrieves information about an override to an on-call rotation</td>
</tr>
<tr>
<td>list_contact_channels</td>
<td>Lists all contact channels for the specified contact</td>
</tr>
<tr>
<td>list_contacts</td>
<td>Lists all contacts and escalation plans in Incident Manager</td>
</tr>
</tbody>
</table>
### Examples

```r
## Not run:
svc <- ssmcontacts()
svc$accept_page(
  Foo = 123
)
## End(Not run)
```

---

**ssmincidents**  
*AWS Systems Manager Incident Manager*

---

**Description**

Systems Manager Incident Manager is an incident management console designed to help users mitigate and recover from incidents affecting their Amazon Web Services-hosted applications. An incident is any unplanned interruption or reduction in quality of services.

Incident Manager increases incident resolution by notifying responders of impact, highlighting relevant troubleshooting data, and providing collaboration tools to get services back up and running. To achieve the primary goal of reducing the time-to-resolution of critical incidents, Incident Manager automates response plans and enables responder team escalation.
Usage

```r
ssmincidents(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**: 
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoint.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoint.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**: 
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
    - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

- **endpoint**: Optional shorthand for complete URL to use for the constructed client.

- **region**: Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```r
svc <- ssmincidents(
  config = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string",
  close_connection = "logical",
  timeout = "numeric",
  s3_force_path_style = "logical",
  sts_regional_endpoint = "string"
),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- **batch_get_incident_findings**: Retrieves details about all specified findings for an incident, including descriptive details about each finding.
- **create_replication_set**: A replication set replicates and encrypts your data to the provided Regions with the provided KMS key.
- **create_response_plan**: Creates a response plan that automates the initial response to incidents.
- **create_timeline_event**: Creates a custom timeline event on the incident details page of an incident record.
- **delete_incident_record**: Delete an incident record from Incident Manager.
- **delete_replication_set**: Deletes all Regions in your replication set.
- **delete_resource_policy**: Deletes the resource policy that Resource Access Manager uses to share your Incident Manager resource.
- **delete_response_plan**: Deletes the specified response plan.
- **delete_timeline_event**: Deletes a timeline event from an incident.
- **get_incident_record**: Returns the details for the specified incident record.
- **get_replication_set**: Retrieve your Incident Manager replication set.
- **get_resource_policies**: Retrieves the resource policies attached to the specified response plan.
- **get_response_plan**: Retrieves the details of the specified response plan.
- **get_timeline_event**: Retrieves a timeline event based on its ID and incident record.
### Description

This API reference provides descriptions, syntax, and other details about each of the actions and data types for AWS Systems Manager for SAP. The topic for each action shows the API request parameters and responses.

### Usage

```r
ssmsap(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```
ssmsap

Arguments

config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- ssmsap(
    config = list(
        credentials = list(
            creds = list(
                access_key_id = "string",
                secret_access_key = "string",
                session_token = "string",
            ),
            profile = "string",
        ),
        anonymous = FALSE,
    ),
    endpoint = "string",
    region = "string",
    close_connection = TRUE,
    timeout = 60,
    s3_force_path_style = FALSE,
    sts_regional_endpoint = "string",
)
```

secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
),
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

delete_resource_permission  Removes permissions associated with the target database
deregister_application     Deregister an SAP application with AWS Systems Manager for SAP
get_application            Gets an application registered with AWS Systems Manager for SAP
get_component              Gets the component of an application registered with AWS Systems Manager for SAP
get_database               Gets the SAP HANA database of an application registered with AWS Systems Manager for SAP
get_operation              Gets the details of an operation by specifying the operation ID
get_resource_permission     Gets permissions associated with the target database
list_applications           Lists all the applications registered with AWS Systems Manager for SAP
list_components             Lists all the components registered with AWS Systems Manager for SAP
list_databases              Lists the SAP HANA databases of an application registered with AWS Systems Manager for SAP
list_operations             Lists the operations performed by AWS Systems Manager for SAP
list_tags_for_resource      Lists all tags on an SAP HANA application and/or database registered with AWS Systems Manager for SAP
put_resource_permission     Adds permissions to the target database
register_application       Register an SAP application with AWS Systems Manager for SAP
start_application_refresh   Refreshes a registered application
tag_resource               Creates tag for a resource by specifying the ARN
untag_resource              Delete the tags for a resource
update_application_settings Updates the settings of an application registered with AWS Systems Manager for SAP
Examples

```r
## Not run:
svc <- ssmsap()
svc$delete_resource_permission(
    Foo = 123
)
## End(Not run)
```

---

**AWS Single Sign-On**

**Description**

AWS IAM Identity Center (successor to AWS Single Sign-On) Portal is a web service that makes it easy for you to assign user access to IAM Identity Center resources such as the AWS access portal. Users can get AWS account applications and roles assigned to them and get federated into the application.

Although AWS Single Sign-On was renamed, the sso and identitystore API namespaces will continue to retain their original name for backward compatibility purposes. For more information, see IAM Identity Center rename.

This reference guide describes the IAM Identity Center Portal operations that you can call programmatically and includes detailed information on data types and errors.

AWS provides SDKs that consist of libraries and sample code for various programming languages and platforms, such as Java, Ruby, .Net, iOS, or Android. The SDKs provide a convenient way to create programmatic access to IAM Identity Center and other AWS services. For more information about the AWS SDKs, including how to download and install them, see Tools for Amazon Web Services.

**Usage**

```r
sso(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

**Arguments**

- `config`: Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
    - `creds`:
      * `access_key_id`: AWS access key ID
      * `secret_access_key`: AWS secret access key
      * `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
    - `anonymous`: Set anonymous credentials.
• **endpoint**: The complete URL to use for the constructed client.
• **region**: The AWS Region used in instantiating the client.
• **close_connection**: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

### credentials
Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

### endpoint
Optional shorthand for complete URL to use for the constructed client.

### region
Optional shorthand for AWS Region used in instantiating the client.

### Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

### Service syntax

```r
svc <- sso(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  )
)`
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

- **get_role_credentials**: Returns the STS short-term credentials for a given role name that is assigned to the user
- **list_account_roles**: Lists all roles that are assigned to the user for a given AWS account
- **list_accounts**: Lists all AWS accounts assigned to the user
- **logout**: Removes the locally stored SSO tokens from the client-side cache and sends an API call to the IAM Identity Center service to invalidate the corresponding server-side IAM Identity Center sign in session

Examples

```r
## Not run:
svc <- sso()
svc$get_role_credentials(
    Foo = 123
)
## End(Not run)
```

Description

IAM Identity Center (successor to Single Sign-On) helps you securely create, or connect, your workforce identities and manage their access centrally across Amazon Web Services accounts and applications. IAM Identity Center is the recommended approach for workforce authentication and authorization in Amazon Web Services, for organizations of any size and type. IAM Identity Center uses the sso and identitystore API namespaces.

This reference guide provides information on single sign-on operations which could be used for access management of Amazon Web Services accounts. For information about IAM Identity Center features, see the IAM Identity Center User Guide.
Many operations in the IAM Identity Center APIs rely on identifiers for users and groups, known as principals. For more information about how to work with principals and principal IDs in IAM Identity Center, see the Identity Store API Reference.

Amazon Web Services provides SDKs that consist of libraries and sample code for various programming languages and platforms (Java, Ruby, .Net, iOS, Android, and more). The SDKs provide a convenient way to create programmatic access to IAM Identity Center and other Amazon Web Services services. For more information about the Amazon Web Services SDKs, including how to download and install them, see Tools for Amazon Web Services.

Usage

```r
ssoadmin(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

- **endpoint**: Optional shorthand for complete URL to use for the constructed client.

- **region**: Optional shorthand for AWS Region used in instantiating the client.
**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- ssoadmin(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

**Operations**

- `attach_customer_managed_policy_reference_to_permission_set` Attaches the specified customer managed policy to the specified permission set.
- `attach_managed_policy_to_permission_set` Attaches an Amazon Web Services managed policy ARN to a permission set.
- `create_account_assignment` Assigns access to a principal for a specified Amazon Web Services account using a specified permission set.
- `create_application` Creates an application in IAM Identity Center for the given Amazon Web Services provider.
- `create_application_assignment` Grant application access to a user or group.
- `create_instance` Creates an instance of IAM Identity Center for a standalone Amazon Web Services account that is not managed by Organizations or a member Amazon Web Services account in an organization.
- `create_instance_access_control_attribute_configuration` Enables the attributes-based access control (ABAC) feature for the specified IAM Identity Center instance.
- `create_permission_set` Creates a permission set within a specified IAM Identity Center instance.
Examples

```r
## Not run:
svc <- ssoadmin()
svc$attach_customer_managed_policy_reference_to_permission_set(
  Foo = 123
)
## End(Not run)
```

Description

IAM Identity Center OpenID Connect (OIDC) is a web service that enables a client (such as CLI or a native application) to register with IAM Identity Center. The service also enables the client to fetch the user’s access token upon successful authentication and authorization with IAM Identity Center.

IAM Identity Center uses the sso and identitystore API namespaces.

Considerations for Using This Guide

Before you begin using this guide, we recommend that you first review the following important information about how the IAM Identity Center OIDC service works.

- The IAM Identity Center OIDC service currently implements only the portions of the OAuth 2.0 Device Authorization Grant standard (https://tools.ietf.org/html/rfc8628) that are necessary to enable single sign-on authentication with the CLI.
• With older versions of the CLI, the service only emits OIDC access tokens, so to obtain a new token, users must explicitly re-authenticate. To access the OIDC flow that supports token refresh and doesn’t require re-authentication, update to the latest CLI version (1.27.10 for CLI V1 and 2.9.0 for CLI V2) with support for OIDC token refresh and configurable IAM Identity Center session durations. For more information, see Configure Amazon Web Services access portal session duration.

• The access tokens provided by this service grant access to all Amazon Web Services account entitlements assigned to an IAM Identity Center user, not just a particular application.

• The documentation in this guide does not describe the mechanism to convert the access token into Amazon Web Services Auth (“sigv4”) credentials for use with IAM-protected Amazon Web Services service endpoints. For more information, see GetRoleCredentials in the IAM Identity Center Portal API Reference Guide.

For general information about IAM Identity Center, see What is IAM Identity Center? in the IAM Identity Center User Guide.

Usage

ssooidc(config = list(), credentials = list(), endpoint = NULL, region = NULL)

Arguments

cconfig Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
• **profile**: The name of a profile to use. If not given, then the default profile is used.
• **anonymous**: Set anonymous credentials.

**endpoint**  
Optional shorthand for complete URL to use for the constructed client.

**region**  
Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- ssooidc(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

**Operations**

- **create_token**  
  Creates and returns access and refresh tokens for clients that are authenticated using client secrets.
**create_token_with_iam**
Creates and returns access and refresh tokens for clients and applications that are authenticated using IAM entities

**register_client**
Registers a client with IAM Identity Center

**start_device_authorization**
Initiates device authorization by requesting a pair of verification codes from the authorization service

---

**Examples**

```r
## Not run:
svc <- ssooidc()
#
svc$create_token(
  clientId = "_yzkThXVzLWhc3QtMQEXAMPLECLIENTID",
  clientSecret = "VERYLONGSECRETeyJraWQiOiJrZXktMTU2MDA5OSIsImFsZyI6IkhTMzg0In0",
  deviceCode = "yJraWQiOiJrZXktMTU2Njk2ODA4OCIImFsZyI6IkhTMzIn0EXAMPLEDEVICECODE",
  grantType = "urn:ietf:params:oauth:grant-type:device-code"
)

## End(Not run)
```

---

**storagegateway**

AWS Storage Gateway

---

**Description**

Storage Gateway Service

Storage Gateway is the service that connects an on-premises software appliance with cloud-based storage to provide seamless and secure integration between an organization’s on-premises IT environment and the Amazon Web Services storage infrastructure. The service enables you to securely upload data to the Amazon Web Services Cloud for cost effective backup and rapid disaster recovery.

Use the following links to get started using the **Storage Gateway Service API Reference**:

- **Storage Gateway required request headers**: Describes the required headers that you must send with every POST request to Storage Gateway.
- **Signing requests**: Storage Gateway requires that you authenticate every request you send; this topic describes how sign such a request.
- **Error responses**: Provides reference information about Storage Gateway errors.
- **Operations in Storage Gateway**: Contains detailed descriptions of all Storage Gateway operations, their request parameters, response elements, possible errors, and examples of requests and responses.
- **Storage Gateway endpoints and quotas**: Provides a list of each Amazon Web Services Region and the endpoints available for use with Storage Gateway.
Storage Gateway resource IDs are in uppercase. When you use these resource IDs with the Amazon EC2 API, EC2 expects resource IDs in lowercase. You must change your resource ID to lowercase to use it with the EC2 API. For example, in Storage Gateway the ID for a volume might be `vol-AA22BB012345DAF670`. When you use this ID with the EC2 API, you must change it to `vol-aa22bb012345daf670`. Otherwise, the EC2 API might not behave as expected.

IDs for Storage Gateway volumes and Amazon EBS snapshots created from gateway volumes are changing to a longer format. Starting in December 2016, all new volumes and snapshots will be created with a 17-character string. Starting in April 2016, you will be able to use these longer IDs so you can test your systems with the new format. For more information, see Longer EC2 and EBS resource IDs.

For example, a volume Amazon Resource Name (ARN) with the longer volume ID format looks like the following:

```
```

A snapshot ID with the longer ID format looks like the following: `snap-78e22663445566ee`.

For more information, see Announcement: Heads-up – Longer Storage Gateway volume and snapshot IDs coming in 2016.

### Usage

```
storagegateway(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

### Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**: 
    - `creds`:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
    - `anonymous`: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`. 
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials** Optional credentials shorthand for the config parameter

• **creds**:
  – **access_key_id**: AWS access key ID
  – **secret_access_key**: AWS secret access key
  – **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

**endpoint** Optional shorthand for complete URL to use for the constructed client.

**region** Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- storagegateway(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
```
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

activate_gateway
add_cache
add_tags_to_resource
add_upload_buffer
add_working_storage
assign_tape_pool
associate_file_system
attach_volume
cancel_archival
cancel_retrieval
create_cachedi_scsci_volume
create_nfs_file_share
create_smb_file_share
create_snapshot
create_snapshot_from_volume_recovery_point
create_storedi_scsci_volume
create_tape_pool
create_tapes
create_tape_with_barcode
delete_automatic_tape_creation_policy
delete_bandwidth_rate_limit
delete_chap_credentials
delete_file_share
delete_gateway
delete_snapshot_schedule
delete_tape
delete_tape_archive
delete_tape_pool
delete_volume
describe_availability_monitor_test
describe_bandwidth_rate_limit
describe_bandwidth_rate_limit_schedule
describe_cache
describe_cachedi_scsci_volumes
describe_chap_credentials
describe_file_system_associations
describe_gateway_information
describe_maintenance_start_time
describe_nfs_file_shares
describe_smb_file_shares

Activates the gateway you previously deployed on your host
Configures one or more gateway local disks as cache for a gateway
Adds one or more tags to the specified resource
Configures one or more gateway local disks as upload buffer for a specified gateway
Configures one or more gateway local disks as working storage for a gateway
Assigns a tape to a tape pool for archiving
Associate an Amazon FSx file system with the FSx File Gateway
Connects a volume to an iSCSI connection and then attaches the volume to the specified gateway
Cancels archiving of a virtual tape to the virtual tape shelf (VTS) after the archiving process is initiated
Cancels retrieval of a virtual tape from the virtual tape shelf (VTS) to a gateway
Creates a cached volume on a specified cached volume gateway
Creates a Network File System (NFS) file share on an existing S3 File Gateway
Creates a Server Message Block (SMB) file share on an existing S3 File Gateway
Initiates a snapshot of a volume
Initiates a snapshot of a gateway from a volume recovery point
Creates a volume on a specified gateway
Creates a new custom tape pool
Creates one or more virtual tapes
Creates a virtual tape by using your own barcode
Deletes the automatic tape creation policy of a gateway
Deletes the bandwidth rate limits of a gateway
Deletes Challenge-Handshake Authentication Protocol (CHAP) credentials for a specified iSCSI target and initiator pair
Deletes a file share from an S3 File Gateway
Deletes a gateway
Deletes a snapshot of a volume
Deletes the specified virtual tape
Deletes the specified virtual tape from the virtual tape shelf (VTS)
Delete a custom tape pool
Deletes the specified storage volume that you previously created using the CreateCachediSCSIVolume or CreateStoragediSCSIVolume API
Returns information about the most recent high availability monitoring test that was performed on the host in a cluster
Returns the bandwidth rate limits of a gateway
Returns information about the bandwidth rate limit schedule of a gateway
Returns information about the cache of a gateway
Returns a description of the gateway volumes specified in the request
Returns an array of Challenge-Handshake Authentication Protocol (CHAP) credentials
Gets the file system association information
Returns metadata about a gateway such as its name, network interfaces, time zone, status, and software version
Returns your gateway’s weekly maintenance start time including the day and time of the week
Gets a description for one or more Network File System (NFS) file shares from a S3 File Gateway
Gets a description for one or more Server Message Block (SMB) file shares.
storagegateway

describe_smb_settings
describe_snapshot_schedule
describe_storedi_scsi_volumes
describe_tape_archives
describe_tape_recovery_points
describe_tapes
describe_upload_buffer
describe_vtl_devices
describe_working_storage
detach_volume
disable_gateway
disassociate_file_system
join_domain
list_automatic_tape_creation_policies
list_file_shares
list_file_system_associations
list_gateways
list_local_disks
list_tags_for_resource
list_tape_pools
list_tapes
list_volume_initiators
list_volume_recovery_points
list_volumes
notify_when_uploaded
refresh_cache
remove_tags_from_resource
reset_cache
retrieve_tape_archive
retrieve_tape_recovery_point
set_local_console_password
set_smb_guest_password
shutdown_gateway
start_availability_monitor_test
start_gateway
update_automatic_tape_creation_policy
update_bandwidth_rate_limit
update_bandwidth_rate_limit_schedule
update_chap_credentials
update_file_system_association
update_gateway_information
update_gateway_software_now
update_maintenance_start_time
update_nfs_file_share
update_smb_file_share
update_smb_file_share_visibility
update_smb_local_groups
update_smb_security_strategy

Gets a description of a Server Message Block (SMB) file share settings from a file gateway
Describes the snapshot schedule for the specified gateway volume
Returns the description of the gateway volumes specified in the request
Returns a description of specified virtual tapes in the virtual tape shelf (VTS)
Returns a list of virtual tape recovery points that are available for the specified virtual tape
Returns a description of virtual tapes that correspond to the specified Amazon Resource Names (ARNs)
Returns information about the upload buffer of a gateway
Returns a description of virtual tape library (VTL) devices for the specified tape volume
Returns information about the working storage of a gateway
Disconnects a volume from an iSCSI connection and then detaches the volume
Disables a tape gateway when the gateway is no longer functioning
Disassociates an Amazon FSx file system from the specified gateway
Adds a file gateway to an Active Directory domain
Lists the automatic tape creation policies for a gateway
Gets a list of the file shares for a specific S3 File Gateway, or the list of file shares that belong to the calling Amazon Web Services account
Gets a list of FileSystemAssociationSummary objects
Lists gateways owned by an Amazon Web Services account in an Amazon Web Services Region
Returns a list of the gateway’s local disks
Lists the tags that have been added to the specified resource
Lists custom tape pools
Lists virtual tapes in your virtual tape library (VTL) and your virtual tape shelf (VTS)
Lists iSCSI initiators that are connected to a volume
Lists the recovery points for a specified gateway
Lists the iSCSI stored volumes of a gateway
Sends you notification through CloudWatch Events when all files written to your virtual tape library have been uploaded to Amazon S3
Refreshes the cached inventory of objects for the specified file share
Removes one or more tags from the specified resource
Resets all cache disks that have encountered an error and makes the disks available for reconfiguration as cache storage
Retrieves an archived virtual tape from the virtual tape shelf (VTS) to a tape gateway
Retrieves the recovery point for the specified virtual tape
Sets the password for your VM local console
Sets the password for the guest user smbguest
Shuts down a Tape Gateway or Volume Gateway
Start a test that verifies that the specified gateway is configured for High Availability
Starts a gateway that you previously shut down (see ShutdownGateway)
Updates the automatic tape creation policy of a gateway
Updates the bandwidth rate limits of a gateway
Updates the bandwidth rate limit schedule for a specified gateway
Updates the Challenge-Handshake Authentication Protocol (CHAP) credentials
Updates a file system association
Updates a gateway’s metadata, which includes the gateway’s name and time zone
Updates the gateway virtual machine (VM) software
Updates a gateway’s weekly maintenance start time information, including day and time
Updates a Network File System (NFS) file share
Updates a Server Message Block (SMB) file share
Controls whether the shares on an S3 File Gateway are visible in a net view
Updates the list of Active Directory users and groups that have special permissions for the gateway
Updates the SMB security strategy on a file gateway
### Not run:

```r
svc <- storagegateway()
# Activates the gateway you previously deployed on your host.
svc$activate_gateway(
    ActivationKey = "29AV1-3OFV9-VVIUB-NKT0I-LRO6V",
    GatewayName = "My_Gateway",
    GatewayRegion = "us-east-1",
    GatewayTimezone = "GMT-12:00",
    GatewayType = "STORED",
    MediumChangerType = "AWS-Gateway-VTL",
    TapeDriveType = "IBM-ULT3580-TD5"
)
```

## End(Not run)

---

### Description

Security Token Service

Security Token Service (STS) enables you to request temporary, limited-privilege credentials for users. This guide provides descriptions of the STS API. For more information about using this service, see [Temporary Security Credentials](#).

### Usage

```r
sts(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

### Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous:** Set anonymous credentials.
- **endpoint:** The complete URL to use for the constructed client.
- **region:** The AWS Region used in instantiating the client.
- **close_connection:** Immediately close all HTTP connections.
- **timeout:** The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style:** Set this to true to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint:** Set sts regional endpoint resolver to regional or legacy `https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html`

**credentials**  
Optional credentials shorthand for the config parameter

- **creds:**
  - **access_key_id:** AWS access key ID
  - **secret_access_key:** AWS secret access key
  - **session_token:** AWS temporary session token
- **profile:** The name of a profile to use. If not given, then the default profile is used.
- **anonymous:** Set anonymous credentials.

**endpoint**  
Optional shorthand for complete URL to use for the constructed client.

**region**  
Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...),` where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- sts(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string",
      ),
      profile = "string",
      anonymous = "logical",
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
  ),
```
sts

sts_regional_endpoint = "string"
),
credentials = list(
  creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
  ),
  profile = "string",
  anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

assume_role
assume_role_with_saml
assume_role_with_web_identity
decode_authorization_message
get_access_key_info
get_caller_identity
get_federation_token
get_session_token

Examples

## Not run:
svc <- sts()
#
svc$assume_role(
  ExternalId = "123ABC",
  Policy = "{"Version":"2012-10-17","Statement":[{"Sid":"Stmt1","Effect":"A...",
  RoleArn = "arn:aws:iam::123456789012:role/demo",
  RoleSessionName = "testAssumeRoleSession",
  Tags = list(
    list(
      Key = "Project",
      Value = "Unicorn"
    ),
    list(
      Key = "Team",
      Value = "Automation"
    ),
    list(
      Key = "Cost-Center",
      Value = "12345"
    )
  )
)
TransitiveTagKeys = list(
    "Project",
    "Cost-Center"
)

## End(Not run)

---

**Description**

Amazon Web Services Support

The *Amazon Web Services Support API Reference* is intended for programmers who need detailed information about the Amazon Web Services Support operations and data types. You can use the API to manage your support cases programmatically. The Amazon Web Services Support API uses HTTP methods that return results in JSON format.

- You must have a Business, Enterprise On-Ramp, or Enterprise Support plan to use the Amazon Web Services Support API.
- If you call the Amazon Web Services Support API from an account that doesn’t have a Business, Enterprise On-Ramp, or Enterprise Support plan, the *SubscriptionRequiredException* error message appears. For information about changing your support plan, see *Amazon Web Services Support*.

You can also use the Amazon Web Services Support API to access features for Trusted Advisor. You can return a list of checks and their descriptions, get check results, specify checks to refresh, and get the refresh status of checks.

You can manage your support cases with the following Amazon Web Services Support API operations:

- The *create_case*, *describe_cases*, *describe_attachment*, and *resolve_case* operations create Amazon Web Services Support cases, retrieve information about cases, and resolve cases.
- The *describe_communications*, *add_communication_to_case*, and *add_attachments_to_set* operations retrieve and add communications and attachments to Amazon Web Services Support cases.
- The *describe_services* and *describe_severity_levels* operations return Amazon Web Service names, service codes, service categories, and problem severity levels. You use these values when you call the *create_case* operation.
You can also use the Amazon Web Services Support API to call the Trusted Advisor operations. For more information, see Trusted Advisor in the Amazon Web Services Support User Guide.

For authentication of requests, Amazon Web Services Support uses Signature Version 4 Signing Process.

For more information about this service and the endpoints to use, see About the Amazon Web Services Support API in the Amazon Web Services Support User Guide.

Usage

```r
support(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

- **endpoint**: Optional shorthand for complete URL to use for the constructed client.

- **region**: Optional shorthand for AWS Region used in instantiating the client.
Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- support(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `add_attachments_to_set`: Adds one or more attachments to an attachment set
- `add_communication_to_case`: Adds additional customer communication to an Amazon Web Services Support case
- `create_case`: Creates a case in the Amazon Web Services Support Center
- `describe_attachment`: Returns the attachment that has the specified ID
- `describe_cases`: Returns a list of cases that you specify by passing one or more case IDs
- `describe_communications`: Returns communications and attachments for one or more support cases
- `describe_create_case_options`: Returns a list of CreateCaseOption types along with the corresponding supported hours and language availability
- `describe_services`: Returns the current list of Amazon Web Services services and a list of service categories.
describe_severity_levels
describe_supported_languages
describe_trusted_advisor_check_refresh_statuses
describe_trusted_advisor_check_result
describe_trusted_advisor_checks
describe_trusted_advisor_check_summaries
refresh_trusted_advisor_check
resolve_case

Returns the list of severity levels that you can assign to a support case
Returns a list of supported languages for a specified categoryCode, issueType
Returns the refresh status of the Trusted Advisor checks that have the specified check IDs
Returns the results of the Trusted Advisor checks that have the specified check IDs
Returns information about all available Trusted Advisor checks, including a list of all the check IDs
Returns the results for the Trusted Advisor check summaries for the check IDs that you specify
Refreshes the Trusted Advisor check that you specify using the check ID
Resolves a support case

Examples

```r
## Not run:
svc <- support()
svc$add_attachments_to_set(Foo = 123)

## End(Not run)
```

---

**Description**

Amazon Web Services Support App in Slack

You can use the Amazon Web Services Support App in Slack API to manage your support cases in Slack for your Amazon Web Services account. After you configure your Slack workspace and channel with the Amazon Web Services Support App, you can perform the following tasks directly in your Slack channel:

- Create, search, update, and resolve your support cases
- Request service quota increases for your account
- Invite Amazon Web Services Support agents to your channel so that you can chat directly about your support cases

For more information about how to perform these actions in Slack, see the following documentation in the *Amazon Web Services Support User Guide*:

- Amazon Web Services Support App in Slack
- Joining a live chat session with Amazon Web Services Support
- Requesting service quota increases
- Amazon Web Services Support App commands in Slack
You can also use the Amazon Web Services Management Console instead of the Amazon Web Services Support App API to manage your Slack configurations. For more information, see Authorize a Slack workspace to enable the Amazon Web Services Support App.

- You must have a Business or Enterprise Support plan to use the Amazon Web Services Support App API.
- For more information about the Amazon Web Services Support App endpoints, see the Amazon Web Services Support App in Slack endpoints in the Amazon Web Services General Reference.

Usage

```r
supportapp(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **creds**:
    - *access_key_id*: AWS access key ID
    - *secret_access_key*: AWS secret access key
    - *session_token*: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest-guide/feature-sts-regionalized-endpoints.html

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - *access_key_id*: AWS access key ID
    - *secret_access_key*: AWS secret access key
    - *session_token*: AWS temporary session token
• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

**endpoint**  
Optional shorthand for complete URL to use for the constructed client.

**region**  
Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- supportapp(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

**Operations**

- **create_slack_channel_configuration**  
  Creates a Slack channel configuration for your Amazon Web Services account
### Examples

```r
## Not run:
svc <- supportapp()
svc$supportapp$create_slack_channel_configuration(
  Foo = 123
)

## End(Not run)
```

---

**Amazon Simple Workflow Service**

**Description**

The Amazon Simple Workflow Service (Amazon SWF) makes it easy to build applications that use Amazon’s cloud to coordinate work across distributed components. In Amazon SWF, a *task* represents a logical unit of work that is performed by a component of your workflow. Coordinating tasks in a workflow involves managing intertask dependencies, scheduling, and concurrency in accordance with the logical flow of the application.

Amazon SWF gives you full control over implementing tasks and coordinating them without worrying about underlying complexities such as tracking their progress and maintaining their state.

This documentation serves as reference only. For a broader overview of the Amazon SWF programming model, see the *Amazon SWF Developer Guide*.

**Usage**

```r
gw(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```
Arguments

config: Optional configuration of credentials, endpoint, and/or region.

- credentials:
  - creds:
    - access_key_id: AWS access key ID
    - secret_access_key: AWS secret access key
    - session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.

- endpoint: The complete URL to use for the constructed client.
- region: The AWS Region used in instantiating the client.
- close_connection: Immediately close all HTTP connections.
- timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
- sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html

credentials: Optional credentials shorthand for the config parameter

- creds:
  - access_key_id: AWS access key ID
  - secret_access_key: AWS secret access key
  - session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.

endpoint: Optional shorthand for complete URL to use for the constructed client.

region: Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- swf(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string",
        profile = "string",
      
```
secret_access_key = "string",
session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string",
close_connection = "logical",
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
  creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
  ),
  profile = "string",
anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)

Operations

<table>
<thead>
<tr>
<th>Operation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>count_closed_workflow_executions</td>
<td>Returns the number of closed workflow executions within the given domain that meet the filtering criteria</td>
</tr>
<tr>
<td>count_open_workflow_executions</td>
<td>Returns the number of open workflow executions within the given domain that meet the filtering criteria</td>
</tr>
<tr>
<td>count_pending_activity_tasks</td>
<td>Returns the estimated number of activity tasks in the specified task list.</td>
</tr>
<tr>
<td>count_pending_decision_tasks</td>
<td>Returns the estimated number of decision tasks in the specified task list.</td>
</tr>
<tr>
<td>deprecate_activity_type</td>
<td>Deprecates the specified activity type.</td>
</tr>
<tr>
<td>deprecate_domain</td>
<td>Deprecates the specified domain.</td>
</tr>
<tr>
<td>deprecate_workflow_type</td>
<td>Deprecates the specified workflow type.</td>
</tr>
<tr>
<td>describe_activity_type</td>
<td>Returns information about the specified activity type.</td>
</tr>
<tr>
<td>describe_domain</td>
<td>Returns information about the specified domain, including description and status.</td>
</tr>
<tr>
<td>describe_workflow_type</td>
<td>Returns information about the specified workflow type.</td>
</tr>
<tr>
<td>describe_workflow_execution</td>
<td>Returns the history of the specified workflow execution.</td>
</tr>
<tr>
<td>get_workflow_execution_history</td>
<td>Returns information about all activities registered in the specified domain that match the filtering criteria.</td>
</tr>
<tr>
<td>list_activity_types</td>
<td>Returns a list of all activities registered in the specified domain.</td>
</tr>
<tr>
<td>list_closed_workflow_executions</td>
<td>Returns a list of closed workflow executions in the specified domain that meet the filtering criteria</td>
</tr>
<tr>
<td>list_domains</td>
<td>Returns a list of all domains registered in the account.</td>
</tr>
<tr>
<td>list_open_workflow_executions</td>
<td>Returns a list of open workflow executions in the specified domain that meet the filtering criteria.</td>
</tr>
<tr>
<td>list_tags_for_resource</td>
<td>List tags for a given domain.</td>
</tr>
<tr>
<td>list_workflow_types</td>
<td>Returns information about workflow types in the specified domain.</td>
</tr>
<tr>
<td>poll_for_activity_task</td>
<td>Used by workers to get an ActivityTask from the specified activity taskList.</td>
</tr>
<tr>
<td>poll_for_decision_task</td>
<td>Used by deciders to get a DecisionTask from the specified decision taskList</td>
</tr>
</tbody>
</table>
Amazon CloudWatch Synthetics

You can use Amazon CloudWatch Synthetics to continually monitor your services. You can create and manage canaries, which are modular, lightweight scripts that monitor your endpoints and APIs from the outside-in. You can set up your canaries to run 24 hours a day, once per minute. The canaries help you check the availability and latency of your web services and troubleshoot anomalies by investigating load time data, screenshots of the UI, logs, and metrics. The canaries seamlessly integrate with CloudWatch ServiceLens to help you trace the causes of impacted nodes in your applications. For more information, see Using ServiceLens to Monitor the Health of Your Applications in the Amazon CloudWatch User Guide.

Before you create and manage canaries, be aware of the security considerations. For more information, see Security Considerations for Synthetics Canaries.
Usage

synthetics(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **creds**:
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
    - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.
- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
    - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

- **endpoint**: Optional shorthand for complete URL to use for the constructed client.
- **region**: Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.
### Service syntax

```r
svc <- synthetics(
  config = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string",
  close_connection = "logical",
  timeout = "numeric",
  s3_force_path_style = "logical",
  sts_regional_endpoint = "string"
),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

### Operations

- **associate_resource**: Associates a canary with a group
- **create_canary**: Creates a canary
- **create_group**: Creates a group which you can use to associate canaries with each other, including cross-Region canaries
- **delete_canary**: Permanently deletes the specified canary
- **delete_group**: Deletes a group
- **describe_canaries**: This operation returns a list of the canaries in your account, along with full details about each canary
- **describe_canaries_last_run**: Use this operation to see information from the most recent run of each canary that you have created
- **describe_runtime_versions**: Returns a list of Synthetics canary runtime versions
- **disassociate_resource**: Removes a canary from a group
- **get_canary**: Retrieves complete information about one canary
- **get_canary_runs**: Retrieves a list of runs for a specified canary
- **get_group**: Returns information about one group
- **list_associated_groups**: Returns a list of the groups that the specified canary is associated with
- **list_group_resources**: This operation returns a list of the ARNs of the canaries that are associated with the specified group
telconetworkbuilder

list_groups
list_tags_for_resource
start_canary
stop_canary
tag_resource
untag_resource
update_canary

Returns a list of all groups in the account, displaying their names, unique IDs, and ARNs
Displays the tags associated with a canary or group
Use this operation to run a canary that has already been created
Stops the canary to prevent all future runs
Assigns one or more tags (key-value pairs) to the specified canary or group
Removes one or more tags from the specified resource
Updates the configuration of a canary that has already been created

Examples

```r
## Not run:
svc <- synthetics()
svc$associate_resource(
  Foo = 123
)

## End(Not run)
```

telconetworkbuilder   AWS Telco Network Builder

Description

Amazon Web Services Telco Network Builder (TNB) is a network automation service that helps you deploy and manage telecom networks. AWS TNB helps you with the lifecycle management of your telecommunication network functions throughout planning, deployment, and post-deployment activities.

Usage

telconetworkbuilder(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)

Arguments

config   Optional configuration of credentials, endpoint, and/or region.

  • credentials:
    • creds:
      • access_key_id: AWS access key ID
      • secret_access_key: AWS secret access key
* session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.

* endpoint: The complete URL to use for the constructed client.
* region: The AWS Region used in instantiating the client.
* close_connection: Immediately close all HTTP connections.
* timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
* s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
* sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter
  - creds:
    - access_key_id: AWS access key ID
    - secret_access_key: AWS secret access key
    - session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.
region Optional shorthand for AWS Region used in instantiating the client.

Value
A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax
```
svc <- telconetworkbuilder(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
```
close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
),
 credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
 endpoint = "string",
 region = "string"
)

Operations

cancel_sol_network_operation  Cancels a network operation
create_sol_function_package  Creates a function package
create_sol_network_instance  Creates a network instance
create_sol_network_package  Creates a network package
delete_sol_function_package  Deletes a function package
delete_sol_network_instance  Deletes a network instance
delete_sol_network_package  Deletes network package
get_sol_function_instance    Gets the details of a network function instance, including the instantiation state and metadata from the function package descriptor in the network function package
get_sol_function_package     Gets the details of an individual function package, such as the operational state and whether it is in use
get_sol_function_package_content Gets the contents of a function package
get_sol_function_package_descriptor Gets a function package descriptor in a function package
get_sol_network_instance     Gets the details of the network instance
get_sol_network_operation    Gets the details of a network operation, including the tasks involved in the network operation and the status of the tasks
get_sol_network_package      Gets the details of a network package
get_sol_network_package_content Gets the contents of a network package
get_sol_network_package_descriptor Gets the content of the network service descriptor
instantiate_sol_network_instance Instantiates a network instance
list_sol_function_instances  Lists network function instances
list_sol_function_packages   Lists information about function packages
list_sol_network_instances   Lists your network instances
list_sol_network_operations  Lists details for a network operation, including when the operation started and the status of the tasks
list_sol_network_packages    Lists network packages
list_tags_for_resource       Lists tags for AWS TNB resources
put_sol_function_package_content Uploads the contents of a function package
put_sol_network_package_content Uploads the contents of a network package
tag_resource                Tags an AWS TNB resource
terminate_sol_network_instance Terminates a network instance
untag_resource               Untags an AWS TNB resource
update_sol_function_package  Updates the operational state of function package
update_sol_network_instance Update a network instance
update_sol_network_package Updates the operational state of a network package
validate_sol_function_package_content Validates function package content
validate_sol_network_package_content Validates network package content

Examples

```r
## Not run:
svc <- telconetworkbuilder()
svc$cancel_sol_network_operation(
   Foo = 123
)
## End(Not run)
```

### Description

Amazon Textract detects and analyzes text in documents and converts it into machine-readable text. This is the API reference documentation for Amazon Textract.

### Usage

textract(config = list(), credentials = list(), endpoint = NULL, region = NULL)

### Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - **access_key_id**: AWS access key ID
      - **secret_access_key**: AWS secret access key
      - **session_token**: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

• **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

**credentials**

Optional credentials shorthand for the `config` parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token

- **profile**: The name of a profile to use. If not given, then the default profile is used.

- **anonymous**: Set anonymous credentials.

**endpoint**

Optional shorthand for complete URL to use for the constructed client.

**region**

Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the `Operations` section.

**Service syntax**

```r
svc <- textract(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string",
  close_connection = "logical",
  timeout = "numeric",
  s3_force_path_style = "logical",
  sts_regional_endpoint = "string"
)
```
secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

analyze_document
analyze_expense
analyze_id
create_adapter
create_adapter_version
delete_adapter
delete_adapter_version
detect_document_text
get_adapter
get_adapter_version
get_document_analysis
get_document_text_detection
get_expense_analysis
get_lending_analysis
get_lending_analysis_summary
list_adapters
list_adapter_versions
list_tags_for_resource
start_document_analysis
start_document_text_detection
start_expense_analysis
start_lending_analysis
tag_resource
untag_resource
update_adapter

Analyzes an input document for relationships between detected items
AnalyzeExpense synchronously analyzes an input document for financially related relationships
Analyzes identity documents for relevant information
Creates an adapter, which can be fine-tuned for enhanced performance on user provided documents
Creates a new version of an adapter
Deletes an Amazon Textract adapter
Deletes an Amazon Textract adapter version
Detects text in the input document
Gets configuration information for an adapter specified by an AdapterId, returning information on AdapterName, Description, CreationTime, AutoUpdate status, and FeatureTypes
Gets configuration information for the specified adapter version, including: AdapterId, AdapterVersion, FeatureTypes, Status, StatusMessage, DatasetConfig, KMSKeyId, OutputConfig, Tags and EvaluationMetrics
Gets the results for an Amazon Textract asynchronous operation that analyzes text in a document
Gets the results for an Amazon Textract asynchronous operation that detects text in a document
Gets the results for an Amazon Textract asynchronous operation that analyzes invoices and receipts
Gets the results for an Amazon Textract asynchronous operation that analyzes text in a lending document
Gets summarized results for the StartLendingAnalysis operation, which analyzes text in a lending document
Lists all adapters that match the specified filtration criteria
Lists all version of an adapter that meet the specified filtration criteria
Lists all tags for an Amazon Textract resource
Starts the asynchronous analysis of an input document for relationships between detected items
Starts the asynchronous detection of text in a document
Starts the asynchronous analysis of invoices or receipts for data like contact information, items purchased, and vendor names
Starts the classification and analysis of an input document
Adds one or more tags to the specified resource
Removes any tags with the specified keys from the specified resource
Update the configuration for an adapter

Examples

## Not run:
svc <- textract()
svc$analyze_document(
    Foo = 123
)

## End(Not run)
Amazon Timestream Query

Description

Amazon Timestream Query

Usage

timestreamquery(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

config Optional configuration of credentials, endpoint, and/or region.

- credentials:
  - creds:
    - access_key_id: AWS access key ID
    - secret_access_key: AWS secret access key
    - session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.
- endpoint: The complete URL to use for the constructed client.
- region: The AWS Region used in instantiating the client.
- close_connection: Immediately close all HTTP connections.
- timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
- sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter

- creds:
  - access_key_id: AWS access key ID
  - secret_access_key: AWS secret access key
  - session_token: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**
A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- timestreamquery(
  config = list("config"),
  credentials = list("credentials"),
  endpoint = "string",
  region = "string",
  close_connection = "logical",
  timeout = "numeric",
  s3_force_path_style = "logical",
  sts_regional_endpoint = "string"
),
credentials = list("credentials"),
endpoint = "string",
region = "string"
)
```

**Operations**

- `cancel_query` Cancels a query that has been issued
timestreamwrite

### create_scheduled_query
Create a scheduled query that will be run on your behalf at the configured schedule

### delete_scheduled_query
Deletes a given scheduled query

### describe_account_settings
Describes the settings for your account that include the query pricing model and the configured maximum TCU

### describe_endpoints
DescribeEndpoints returns a list of available endpoints to make Timestream API calls against

### describe_scheduled_query
Provides detailed information about a scheduled query

### execute_scheduled_query
You can use this API to run a scheduled query manually

### list_scheduled_queries
Gets a list of all scheduled queries in the caller’s Amazon account and Region

### list_tags_for_resource
List all tags on a Timestream query resource

### prepare_query
A synchronous operation that allows you to submit a query with parameters to be stored by Timestream for later running

### query
Query is a synchronous operation that enables you to run a query against your Amazon Timestream database

### tag_resource
Associate a set of tags with a Timestream resource

### untag_resource
Removes the association of tags from a Timestream query resource

### update_account_settings
Transitions your account to use TCUs for query pricing and modifies the maximum query compute units that you’ve configured

### update_scheduled_query
Update a scheduled query

### Examples

```r
## Not run:
svc <- timestreamquery()
svc$cancel_query(
  Foo = 123
)

## End(Not run)
```

---

timestreamwrite | *Amazon Timestream Write*

### Description

Amazon Timestream is a fast, scalable, fully managed time-series database service that makes it easy to store and analyze trillions of time-series data points per day. With Timestream, you can easily store and analyze IoT sensor data to derive insights from your IoT applications. You can analyze industrial telemetry to streamline equipment management and maintenance. You can also store and analyze log data and metrics to improve the performance and availability of your applications.

Timestream is built from the ground up to effectively ingest, process, and store time-series data. It organizes data to optimize query processing. It automatically scales based on the volume of data ingested and on the query volume to ensure you receive optimal performance while inserting and querying data. As your data grows over time, Timestream’s adaptive query processing engine spans across storage tiers to provide fast analysis while reducing costs.
Usage

timestreamwrite(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)

Arguments

config Optional configuration of credentials, endpoint, and/or region.

  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.

  • endpoint: The complete URL to use for the constructed client.

  • region: The AWS Region used in instantiating the client.

  • close_connection: Immediately close all HTTP connections.

  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy [link](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

credentials Optional credentials shorthand for the config parameter

  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.

epsilon Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```r
timestreamwrite()
svc <- timestreamwrite(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `create_batch_load_task`: Creates a new Timestream batch load task
- `create_database`: Creates a new Timestream database
- `create_table`: Adds a new table to an existing database in your account
- `delete_database`: Deletes a given Timestream database
- `delete_table`: Deletes a given Timestream table
- `describe_batch_load_task`: Returns information about the batch load task, including configurations, mappings, progress, and other details
- `describe_database`: Returns information about the database, including the database name, time that the database was created, and the total number of tables found within the database
- `describe_endpoints`: Returns a list of available endpoints to make Timestream API calls against
- `describe_table`: Returns information about the table, including the table name, database name, retention duration of the memory store and the magnetic store
- `list_batch_load_tasks`: Provides a list of batch load tasks, along with the name, status, when the task is resumable until, and other details
- `list_databases`: Returns a list of your Timestream databases
- `list_tables`: Provides a list of tables, along with the name, status, and retention properties of each table
- `list_tags_for_resource`: Lists all tags on a Timestream resource
- `resume_batch_load_task`: Resume batch load task

```r
timestreamwrite
```
tag_resource Associates a set of tags with a Timestream resource
untag_resource Removes the association of tags from a Timestream resource
update_database Modifies the KMS key for an existing database
update_table Modifies the retention duration of the memory store and magnetic store for your Timestream table
write_records Enables you to write your time-series data into Timestream

Examples

```r
## Not run:
svc <- timestreamwrite()
svc$create_batch_load_task(
  Foo = 123
)
## End(Not run)
```

Description

Amazon Transcribe offers three main types of batch transcription: **Standard**, **Medical**, and **Call Analytics**.

- **Standard transcriptions** are the most common option. Refer to for details.
- **Medical transcriptions** are tailored to medical professionals and incorporate medical terms. A common use case for this service is transcribing doctor-patient dialogue into after-visit notes. Refer to for details.
- **Call Analytics transcriptions** are designed for use with call center audio on two different channels; if you’re looking for insight into customer service calls, use this option. Refer to for details.

Usage

```r
transcribeservice(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```
Arguments

config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/Bucket/Key.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

document

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- transcribeservice(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string",
      ),
      profile = "profile_name",
      anonymous = FALSE,
    ),
    endpoint = "http://example.com",
    region = "us-east-1",
  ),
  close_connection = TRUE,
  timeout = 120,  # seconds
  s3_force_path_style = TRUE,
  sts_regional_endpoint = "legacy",
)
```

transcribeservice

secret_access_key = "string",
session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string",
close_connection = "logical",
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
creds = list(
access_key_id = "string",
secret_access_key = "string",
session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

create_call_analytics_category Creates a new Call Analytics category
create_language_model Creates a new custom language model
create_medical_vocabulary Creates a new custom medical vocabulary
create_vocabulary Creates a new custom vocabulary
create_vocabulary_filter Creates a new custom vocabulary filter
delete_call_analytics_category Deletes a Call Analytics category
delete_call_analytics_job Deletes a Call Analytics job
delete_language_model Deletes a custom language model
delete_medical_scribe_job Deletes a Medical Scribe job
delete_medical_transcription_job Deletes a medical transcription job
delete_medical_vocabulary Deletes a custom medical vocabulary
delete_transcription_job Deletes a transcription job
delete_vocabulary Deletes a custom vocabulary
delete_vocabulary_filter Deletes a custom vocabulary filter
describe_language_model Provides information about the specified custom language model
describe_call_analytics_category Provides information about the specified Call Analytics category
describe_call_analytics_job Provides information about the specified Call Analytics job
describe_medical_scribe_job Provides information about the specified Medical Scribe job
describe_medical_transcription_job Provides information about the specified medical transcription job
describe_medical_vocabulary Provides information about the specified custom medical vocabulary
get_transcription_job  Provides information about the specified transcription job
get_vocabulary  Provides information about the specified custom vocabulary
get_vocabulary_filter  Provides information about the specified custom vocabulary filter
list_call_analytics_categories  Provides a list of Call Analytics categories, including all rules that make up each category
list_call_analytics_jobs  Provides a list of Call Analytics jobs that match the specified criteria
list_language_models  Provides a list of custom language models that match the specified criteria
list_medical_scribe_jobs  Provides a list of Medical Scribe jobs that match the specified criteria
list_medical_transcription_jobs  Provides a list of medical transcription jobs that match the specified criteria
list_medical_vocabularies  Provides a list of custom medical vocabularies that match the specified criteria
list_tags_for_resource  Lists all tags associated with the specified transcription job, vocabulary, model, or resource
list_transcription_jobs  Provides a list of transcription jobs that match the specified criteria
list_vocabularies  Provides a list of custom vocabularies that match the specified criteria
list_vocabulary_filters  Provides a list of custom vocabulary filters that match the specified criteria
start_call_analytics_job  Transcribes the audio from a customer service call and applies any additional Request Parameters you choose to include in your request
start_medical_scribe_job  Transcribes patient-clinician conversations and generates clinical notes
start_medical_transcription_job  Transcribes the audio from a medical dictation or conversation and applies any additional Request Parameters you choose to include in your request
tag_resource  Adds one or more custom tags, each in the form of a key:value pair, to the specified resource
untag_resource  Removes the specified tags from the specified Amazon Transcribe resource
update_call_analytics_category  Updates the specified Call Analytics category with new rules
update_medical_vocabulary  Updates an existing custom medical vocabulary with new values
update_vocabulary  Updates an existing custom vocabulary with new values
update_vocabulary_filter  Updates an existing custom vocabulary filter with a new list of words

## Examples

```r
## Not run:
svc <- transcribeservice()
svc$create_call_analytics_category(
  Foo = 123
)

## End(Not run)
```

### Description

Provides translation of the input content from the source language to the target language.
Usage

```
translate(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **creds**:
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
    - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.
- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
    - **session_token**: AWS temporary session token
  - **profile**: The name of a profile to use. If not given, then the default profile is used.
  - **anonymous**: Set anonymous credentials.

- **endpoint**: Optional shorthand for complete URL to use for the constructed client.
- **region**: Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```
svc <- translate(
    config = list(
        credentials = list(
            creds = list(
                access_key_id = "string",
                secret_access_key = "string",
                session_token = "string"
            ),
            profile = "string",
            anonymous = "logical"
        ),
        endpoint = "string",
        region = "string",
        close_connection = "logical",
        timeout = "numeric",
        s3_force_path_style = "logical",
        sts_regional_endpoint = "string"
    ),
    credentials = list(
        creds = list(
            access_key_id = "string",
            secret_access_key = "string",
            session_token = "string"
        ),
        profile = "string",
        anonymous = "logical"
    ),
    endpoint = "string",
    region = "string"
)
```

Operations

- `create_parallel_data`: Creates a parallel data resource in Amazon Translate by importing an input file from Amazon S3
- `delete_parallel_data`: Deletes a parallel data resource in Amazon Translate
- `delete_terminology`: A synchronous action that deletes a custom terminology
- `describe_text_translation_job`: Gets the properties associated with an asynchronous batch translation job including name, ID, status, source and target languages, input/output S3 buckets, and so on
- `get_parallel_data`: Provides information about a parallel data resource
- `get_terminology`: Retrieves a custom terminology
- `import_terminology`: Creates or updates a custom terminology, depending on whether one already exists for the given terminology name
- `list_languages`: Provides a list of languages (RFC-5646 codes and names) that Amazon Translate supports
- `list_parallel_data`: Provides a list of your parallel data resources in Amazon Translate
- `list_tags_for_resource`: Lists all tags associated with a given Amazon Translate resource
- `list_terminologies`: Provides a list of custom terminologies associated with your account
- `list_text_translation_jobs`: Gets a list of the batch translation jobs that you have submitted
- `start_text_translation_job`: Starts an asynchronous batch translation job
- `stop_text_translation_job`: Stops an asynchronous batch translation job that is in progress
verifiedpermissions

| tag_resource | Associates a specific tag with a resource |
| translate_document | Translates the input document from the source language to the target language |
| translate_text | Translates input text from the source language to the target language |
| untag_resource | Removes a specific tag associated with an Amazon Translate resource |
| update_parallel_data | Updates a previously created parallel data resource by importing a new input file from Amazon |

Examples

```r
## Not run:
svc <- translate()
svc$create_parallel_data(
  Foo = 123
)

## End(Not run)
```

---

**verifiedpermissions**  
*Amazon Verified Permissions*

**Description**

Amazon Verified Permissions is a permissions management service from Amazon Web Services. You can use Verified Permissions to manage permissions for your application, and authorize user access based on those permissions. Using Verified Permissions, application developers can grant access based on information about the users, resources, and requested actions. You can also evaluate additional information like group membership, attributes of the resources, and session context, such as time of request and IP addresses. Verified Permissions manages these permissions by letting you create and store authorization policies for your applications, such as consumer-facing web sites and enterprise business systems.

Verified Permissions uses Cedar as the policy language to express your permission requirements. Cedar supports both role-based access control (RBAC) and attribute-based access control (ABAC) authorization models.

For more information about configuring, administering, and using Amazon Verified Permissions in your applications, see the [Amazon Verified Permissions User Guide](https://aws.amazon.com/verified-permissions/).

For more information about the Cedar policy language, see the [Cedar Policy Language Guide](https://awslabs.github.io/Cedar/language/).

When you write Cedar policies that reference principals, resources and actions, you can define the unique identifiers used for each of those elements. We strongly recommend that you follow these best practices:

- **Use values like universally unique identifiers (UUIDs) for all principal and resource identifiers.**

  For example, if user *jane* leaves the company, and you later let someone else use the name *jane*, then that new user automatically gets access to everything granted by policies that still
reference User::"jane". Cedar can’t distinguish between the new user and the old. This applies to both principal and resource identifiers. Always use identifiers that are guaranteed unique and never reused to ensure that you don’t unintentionally grant access because of the presence of an old identifier in a policy.

Where you use a UUID for an entity, we recommend that you follow it with the // comment specifier and the ‘friendly’ name of your entity. This helps to make your policies easier to understand. For example: principal == User::"a1b2c3d4-e5f6-a1b2-c3d4-EXAMPLE11111", // alice

- **Do not include personally identifying, confidential, or sensitive information as part of the unique identifier for your principals or resources.** These identifiers are included in log entries shared in CloudTrail trails.

Several operations return structures that appear similar, but have different purposes. As new functionality is added to the product, the structure used in a parameter of one operation might need to change in a way that wouldn’t make sense for the same parameter in a different operation. To help you understand the purpose of each, the following naming convention is used for the structures:

- Parameter type structures that end in `Detail` are used in `Get` operations.
- Parameter type structures that end in `Item` are used in `List` operations.
- Parameter type structures that use neither suffix are used in the mutating (create and update) operations.

### Usage

```
verifiedpermissions(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

### Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**:
    - **creds**:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - **profile**: The name of a profile to use. If not given, then the default profile is used.
    - **anonymous**: Set anonymous credentials.
  - **endpoint**: The complete URL to use for the constructed client.
  - **region**: The AWS Region used in instantiating the client.
  - **close_connection**: Immediately close all HTTP connections.
  - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
### credentials

Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

### endpoint

Optional shorthand for complete URL to use for the constructed client.

### region

Optional shorthand for AWS Region used in instantiating the client.

### Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

### Service syntax

```r
svc <- verifiedpermissions(
  config = list(  
    credentials = list(  
      creds = list(  
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),  
      profile = "string",
      anonymous = "logical"  
    ),  
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(  
    creds = list(  
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    )
  )
)
```
Operations

- **batch_is_authorized**
  Makes a series of decisions about multiple authorization requests for one principal or resource

- **batch_is_authorized_with_token**
  Makes a series of decisions about multiple authorization requests for one token

- **create_identity_source**
  Creates a reference to an Amazon Cognito user pool as an external identity provider (IdP)

- **create_policy**
  Creates a Cedar policy and saves it in the specified policy store

- **create_policy_store**
  Creates a policy store

- **create_policy_template**
  Creates a policy template

- **delete_identity_source**
  Deletes an identity source that references an identity provider (IdP) such as Amazon Cognito

- **delete_policy**
  Deletes the specified policy from the policy store

- **delete_policy_store**
  Deletes the specified policy store

- **delete_policy_template**
  Deletes the specified policy template from the policy store

- **get_identity_source**
  Retrieves the details about the specified identity source

- **get_policy**
  Retrieves information about the specified policy

- **get_policy_source**
  Retrieves details about a policy store

- **get_policy_template**
  Retrieves the details for the specified policy template in the specified policy store

- **get_schema**
  Retrieve the details for the specified schema in the specified policy store

- **is_authorized**
  Makes an authorization decision about a service request described in the parameters

- **is_authorized_with_token**
  Makes an authorization decision about a service request described in the parameters

- **list_identity_sources**
  Returns a paginated list of all of the identity sources defined in the specified policy store

- **list_policies**
  Returns a paginated list of all policies stored in the specified policy store

- **list_policy_stores**
  Returns a paginated list of all policy stores in the calling Amazon Web Services account

- **list_policy_templates**
  Returns a paginated list of all policy templates in the specified policy store

- **put_schema**
  Creates or updates the policy schema in the specified policy store

- **update_identity_source**
  Updates the specified identity source to use a new identity provider (IdP) source, or to change the mapping of identities from the IdP to a different principal entity type

- **update_policy**
  Modifies a Cedar static policy in the specified policy store

- **update_policy_store**
  Modifies the validation setting for a policy store

- **update_policy_template**
  Updates the specified policy template

Examples

```r
## Not run:
svc <- verifiedpermissions()
svc$batch_is_authorized(
    Foo = 123
)

## End(Not run)
```
Amazon Voice ID provides real-time caller authentication and fraud risk detection, which make voice interactions in contact centers more secure and efficient.

Usage

voiceid(config = list(), credentials = list(), endpoint = NULL, region = NULL)

Arguments

config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.
Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- voiceid(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `associate_fraudster` Associates the fraudsters with the watchlist specified in the same domain
- `create_domain` Creates a domain that contains all Amazon Connect Voice ID data, such as speakers, fraudsters, audio, and voiceprints
- `create_watchlist` Creates a watchlist that fraudsters can be a part of
- `delete_domain` Deletes the specified domain from Voice ID
- `delete_fraudster` Deletes the specified fraudster from Voice ID
- `delete_speaker` Deletes the specified speaker from Voice ID
- `delete_watchlist` Deletes the specified watchlist from Voice ID
- `describe_domain` Describes the specified domain
<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>describe_fraudster</code></td>
<td>Describes the specified fraudster</td>
</tr>
<tr>
<td><code>describe_fraudster_registration_job</code></td>
<td>Describes the specified fraudster registration job</td>
</tr>
<tr>
<td><code>describe_speaker</code></td>
<td>Describes the specified speaker</td>
</tr>
<tr>
<td><code>describe_speaker_enrollment_job</code></td>
<td>Describes the specified speaker enrollment job</td>
</tr>
<tr>
<td><code>describe_watchlist</code></td>
<td>Describes the specified watchlist</td>
</tr>
<tr>
<td><code>disassociate_fraudster</code></td>
<td>Disassociates the fraudsters from the watchlist specified</td>
</tr>
<tr>
<td><code>evaluate_session</code></td>
<td>Evaluates a specified session based on audio data accumulated during a streaming Amazon Connect Voice ID call</td>
</tr>
<tr>
<td><code>list_domains</code></td>
<td>Lists all the domains in the Amazon Web Services account</td>
</tr>
<tr>
<td><code>list_fraudster_registration_jobs</code></td>
<td>Lists all the fraudster registration jobs in the domain with the given JobStatus</td>
</tr>
<tr>
<td><code>list_fraudsters</code></td>
<td>Lists all fraudsters in a specified watchlist or domain</td>
</tr>
<tr>
<td><code>list_speaker_enrollment_jobs</code></td>
<td>Lists all the speaker enrollment jobs in the domain with the specified JobStatus</td>
</tr>
<tr>
<td><code>list_speakers</code></td>
<td>Lists all speakers in a specified domain</td>
</tr>
<tr>
<td><code>list_tags_for_resource</code></td>
<td>Lists all tags associated with a specified Voice ID resource</td>
</tr>
<tr>
<td><code>list_watchlists</code></td>
<td>Lists all watchlists in a specified domain</td>
</tr>
<tr>
<td><code>opt_out_speaker</code></td>
<td>Opts out a speaker from Voice ID</td>
</tr>
<tr>
<td><code>start_fraudster_registration_job</code></td>
<td>Starts a new batch fraudster registration job using provided details</td>
</tr>
<tr>
<td><code>start_speaker_enrollment_job</code></td>
<td>Starts a new batch speaker enrollment job using specified details</td>
</tr>
<tr>
<td><code>tag_resource</code></td>
<td>Tags a Voice ID resource with the provided list of tags</td>
</tr>
<tr>
<td><code>untag_resource</code></td>
<td>Removes specified tags from a specified Amazon Connect Voice ID resource</td>
</tr>
<tr>
<td><code>update_domain</code></td>
<td>Updates the specified domain</td>
</tr>
<tr>
<td><code>update_watchlist</code></td>
<td>Updates the specified watchlist</td>
</tr>
</tbody>
</table>

**Examples**

```r
## Not run:
svc <- voiceid()
svc$associate_fraudster(
    Foo = 123
)

## End(Not run)
```

---

**vpclattice**

Amazon VPC Lattice

**Description**

Amazon VPC Lattice is a fully managed application networking service that you use to connect, secure, and monitor all of your services across multiple accounts and virtual private clouds (VPCs). Amazon VPC Lattice interconnects your microservices and legacy services within a logical boundary, so that you can discover and manage them more efficiently. For more information, see the Amazon VPC Lattice User Guide
Usage

```r
vpclattice(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>config</td>
<td>Optional configuration of credentials, endpoint, and/or region.</td>
</tr>
<tr>
<td></td>
<td>• <strong>credentials:</strong></td>
</tr>
<tr>
<td></td>
<td>- <strong>creds:</strong></td>
</tr>
<tr>
<td></td>
<td>• <strong>access_key_id:</strong> AWS access key ID</td>
</tr>
<tr>
<td></td>
<td>• <strong>secret_access_key:</strong> AWS secret access key</td>
</tr>
<tr>
<td></td>
<td>• <strong>session_token:</strong> AWS temporary session token</td>
</tr>
<tr>
<td></td>
<td>- <strong>profile:</strong> The name of a profile to use. If not given, then the default</td>
</tr>
<tr>
<td></td>
<td>profile is used.</td>
</tr>
<tr>
<td></td>
<td>- <strong>anonymous:</strong> Set anonymous credentials.</td>
</tr>
<tr>
<td></td>
<td>• <strong>endpoint:</strong> The complete URL to use for the constructed client.</td>
</tr>
<tr>
<td></td>
<td>• <strong>region:</strong> The AWS Region used in instantiating the client.</td>
</tr>
<tr>
<td></td>
<td>• <strong>close_connection:</strong> Immediately close all HTTP connections.</td>
</tr>
<tr>
<td></td>
<td>• <strong>timeout:</strong> The time in seconds till a timeout exception is thrown when</td>
</tr>
<tr>
<td></td>
<td>attempting to make a connection. The default is 60 seconds.</td>
</tr>
<tr>
<td></td>
<td>• <strong>s3_force_path_style:</strong> Set this to true to force the request to use</td>
</tr>
<tr>
<td></td>
<td>path-style addressing, i.e. <a href="http://s3.amazonaws.com/BUCKET/KEY">http://s3.amazonaws.com/BUCKET/KEY</a>.</td>
</tr>
<tr>
<td></td>
<td>• <strong>sts_regional_endpoint:</strong> Set sts regional endpoint resolver to regional</td>
</tr>
<tr>
<td></td>
<td>or legacy <a href="https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-">https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-</a></td>
</tr>
<tr>
<td></td>
<td>regionalized-endpoints.html</td>
</tr>
<tr>
<td>credentials</td>
<td>Optional credentials shorthand for the config parameter</td>
</tr>
<tr>
<td></td>
<td>• <strong>creds:</strong></td>
</tr>
<tr>
<td></td>
<td>- <strong>access_key_id:</strong> AWS access key ID</td>
</tr>
<tr>
<td></td>
<td>- <strong>secret_access_key:</strong> AWS secret access key</td>
</tr>
<tr>
<td></td>
<td>- <strong>session_token:</strong> AWS temporary session token</td>
</tr>
<tr>
<td></td>
<td>• <strong>profile:</strong> The name of a profile to use. If not given, then the default</td>
</tr>
<tr>
<td></td>
<td>profile is used.</td>
</tr>
<tr>
<td></td>
<td>• <strong>anonymous:</strong> Set anonymous credentials.</td>
</tr>
<tr>
<td>endpoint</td>
<td>Optional shorthand for complete URL to use for the constructed client.</td>
</tr>
<tr>
<td>region</td>
<td>Optional shorthand for AWS Region used in instantiating the client.</td>
</tr>
</tbody>
</table>

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...),` where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```r
svc <- vpclattice(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `batch_update_rule`: Updates the listener rules in a batch
- `create_access_log_subscription`: Enables access logs to be sent to Amazon CloudWatch, Amazon S3, and Amazon Kinesis Data Firehose
- `create_listener`: Creates a listener for a service
- `create_rule`: Creates a listener rule
- `create_service`: Creates a service
- `create_service_network`: Creates a service network
- `create_service_network_service_association`: Associates a service with a service network
- `create_service_network_vpc_association`: Associates a VPC with a service network
- `create_target_group`: Creates a target group
- `delete_access_log_subscription`: Deletes the specified access log subscription
- `delete_auth_policy`: Deletes the specified auth policy
- `delete_listener`: Deletes the specified listener
- `delete_resource_policy`: Deletes the specified resource policy
- `delete_rule`: Deletes a listener rule
delete_service
delete_service_network
delete_service_network_service_association
delete_service_network_vpc_association
delete_target_group
deregister_targets
get_access_log_subscription
get_auth_policy
get_listener
get_resource_policy
get_rule
get_service
get_service_network
get_service_network_service_association
get_service_network_vpc_association
get_target_group
list_access_log_subscriptions
list_listeners
list_rules
list_service_networks
list_service_network_service_associations
list_service_network_vpc_associations
list_services
list_tags_for_resource
list_target_groups
list_targets
put_auth_policy
put_resource_policy
register_targets
tag_resource
untag_resource
update_access_log_subscription
update_listener
update_rule
update_service
update_service_network
update_service_network_vpc_association
update_target_group

Examples

## Not run:

svc <- vpclattice()
svc$batch_update_rule(
  Foo = 123
)

Deletes a service
Deletes a service network
Deletes the association between a specified service and the specific service network
Disassociates the VPC from the service network
Deletes a target group
Deregisters the specified targets from the specified target group
Retrieves information about the specified access log subscription
Retrieves information about the auth policy for the specified service or service network
Retrieves information about the specified listener for the specified service
Retrieves information about the resource policy
Retrieves information about listener rules
Retrieves information about the specified service
Retrieves information about the specified service network
Retrieves information about the specified association between a service network and a service
Retrieves information about the association between a service network and a VPC
Retrieves information about the specified target group
Lists all access log subscriptions for the specified service network or service
Lists the listeners for the specified service
Lists the rules for the listener
Lists the service networks owned by the caller account or shared with the caller account
Lists the associations between the service network and the service
Lists the service network and VPC associations
Lists the services owned by the caller account or shared with the caller account
Lists the tags for the specified resource
Lists your target groups
Lists the targets for the target group
Creates or updates the auth policy
Attaches a resource-based permission policy to a service or service network
Registers the targets with the target group
Adds the specified tags to the specified resource
Removes the specified tags from the specified resource
Updates the specified access log subscription
Updates the specified listener for the specified service
Updates a rule for the listener
Updates the specified service
Updates the specified service network
Updates the service network and VPC association
Updates the specified target group
Description

This is AWS WAF Classic documentation. For more information, see AWS WAF Classic in the developer guide.

For the latest version of AWS WAF, use the AWS WAFV2 API and see the AWS WAF Developer Guide. With the latest version, AWS WAF has a single set of endpoints for regional and global use.

This is the AWS WAF Classic API Reference for using AWS WAF Classic with Amazon CloudFront. The AWS WAF Classic actions and data types listed in the reference are available for protecting Amazon CloudFront distributions. You can use these actions and data types via the endpoint waf.amazonaws.com. This guide is for developers who need detailed information about the AWS WAF Classic API actions, data types, and errors. For detailed information about AWS WAF Classic features and an overview of how to use the AWS WAF Classic API, see the AWS WAF Classic in the developer guide.

Usage

waf(config = list(), credentials = list(), endpoint = NULL, region = NULL)

Arguments

config Optional configuration of credentials, endpoint, and/or region.

- credentials:
  - creds:
    * access_key_id: AWS access key ID
    * secret_access_key: AWS secret access key
    * session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.

- endpoint: The complete URL to use for the constructed client.

- region: The AWS Region used in instantiating the client.

- close_connection: Immediately close all HTTP connections.

- timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

- s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

- sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoint.html
credentials  Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile
    is used.
  • anonymous: Set anonymous credentials.

endpoint  Optional shorthand for complete URL to use for the constructed client.

region  Optional shorthand for AWS Region used in instantiating the client.

Value
A client for the service. You can call the service’s operations using syntax like svc$operation(...),
where svc is the name you’ve assigned to the client. The available operations are listed in the Op-
erations section.

Service syntax

```r
svc <- waf(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
)```

region = "string"
)

Operations

create_byte_match_set
create_geo_match_set
create_ip_set
create_rate_based_rule
create_regex_match_set
create_regex_pattern_set
create_rule
create_rule_group
create_size_constraint_set
create_sql_injection_match_set
create_web_acl
create_web_acl_migration_stack
delete_byte_match_set
delete_geo_match_set
delete_ip_set
delete_logging_configuration
delete_permission_policy
delete_rate_based_rule
delete_regex_match_set
delete_regex_pattern_set
delete_rule
delete_rule_group
delete_size_constraint_set
delete_sql_injection_match_set
delete_web_acl
delete_xss_match_set
get_byte_match_set
get_change_token
get_change_token_status
get_geo_match_set
get_ip_set
get_logging_configuration
get_permission_policy
get_rate_based_rule
get_rate_based_rule_managed_keys
get_regex_match_set
get_regex_pattern_set
get_rule
get_rule_group
get_sampled_requests
get_size_constraint_set
get_sql_injection_match_set

This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation
This is AWS WAF Classic documentation

create_web_acl

Creates an AWS CloudFormation WAFV2 template for the specified web ACL in the specified Amazon S3 bucket.
Examples

```r
## Not run:
svc <- waf()
# The following example creates an IP match set named MyIPSetFriendlyName.
svc$create_ip_set(
  ChangeToken = "abcd12f2-46da-4fdb-b8d5-fbd4c466928f",
  Name = "MyIPSetFriendlyName"
)
```

## End(Not run)
Description

This is AWS WAF Classic Regional documentation. For more information, see AWS WAF Classic in the developer guide.

For the latest version of AWS WAF, use the AWS WAFV2 API and see the AWS WAF Developer Guide. With the latest version, AWS WAF has a single set of endpoints for regional and global use.

This is the AWS WAF Regional Classic API Reference for using AWS WAF Classic with the AWS resources, Elastic Load Balancing (ELB) Application Load Balancers and API Gateway APIs. The AWS WAF Classic actions and data types listed in the reference are available for protecting Elastic Load Balancing (ELB) Application Load Balancers and API Gateway APIs. You can use these actions and data types by means of the endpoints listed in AWS Regions and Endpoints. This guide is for developers who need detailed information about the AWS WAF Classic API actions, data types, and errors. For detailed information about AWS WAF Classic features and an overview of how to use the AWS WAF Classic API, see the AWS WAF Classic in the developer guide.

Usage

```python
wafregional(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>config</td>
<td>Optional configuration of credentials, endpoint, and/or region.</td>
</tr>
<tr>
<td>credentials</td>
<td>• credentials:</td>
</tr>
<tr>
<td></td>
<td>• creds:</td>
</tr>
<tr>
<td></td>
<td>• access_key_id: AWS access key ID</td>
</tr>
<tr>
<td></td>
<td>• secret_access_key: AWS secret access key</td>
</tr>
<tr>
<td></td>
<td>• session_token: AWS temporary session token</td>
</tr>
<tr>
<td></td>
<td>• profile: The name of a profile to use. If not given, then the default profile is used.</td>
</tr>
<tr>
<td></td>
<td>• anonymous: Set anonymous credentials.</td>
</tr>
<tr>
<td></td>
<td>• endpoint: The complete URL to use for the constructed client.</td>
</tr>
<tr>
<td></td>
<td>• region: The AWS Region used in instantiating the client.</td>
</tr>
<tr>
<td></td>
<td>• close_connection: Immediately close all HTTP connections.</td>
</tr>
<tr>
<td></td>
<td>• timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.</td>
</tr>
<tr>
<td></td>
<td>• s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. <a href="http://s3.amazonaws.com/BUCKET/KEY">http://s3.amazonaws.com/BUCKET/KEY</a>.</td>
</tr>
</tbody>
</table>
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials**
Optional credentials shorthand for the config parameter

• **creds**:
  – **access_key_id**: AWS access key ID
  – **secret_access_key**: AWS secret access key
  – **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

---

**Value**

A client for the service. You can call the service’s operations using syntax like `svc(operation(...))`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- wafregional(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  timeout = "numeric",
  s3_force_path_style = "logical"
)
```
Anonymous = "logical",
endpoint = "string",
region = "string"
)

Operations

associate_web_acl This is AWS WAF Classic Regional documentation
create_byte_match_set This is AWS WAF Classic documentation
create_geo_match_set This is AWS WAF Classic documentation
create_ip_set This is AWS WAF Classic documentation
create_rate_based_rule This is AWS WAF Classic documentation
create_regex_match_set This is AWS WAF Classic documentation
create_regex_pattern_set This is AWS WAF Classic documentation
create_rule This is AWS WAF Classic documentation
create_rule_group This is AWS WAF Classic documentation
create_size_constraint_set This is AWS WAF Classic documentation
create_sql_injection_match_set This is AWS WAF Classic documentation
create_web_acl This is AWS WAF Classic documentation
create_web_acl_migration_stack Creates an AWS CloudFormation WAFV2 template for the specified web ACL in the specified Amazon S3 bucket
create_xss_match_set This is AWS WAF Classic documentation
delete_byte_match_set This is AWS WAF Classic documentation
delete_geo_match_set This is AWS WAF Classic documentation
delete_ip_set This is AWS WAF Classic documentation
delete_logging_configuration This is AWS WAF Classic documentation
delete_permission_policy This is AWS WAF Classic documentation
delete_rate_based_rule This is AWS WAF Classic documentation
delete_regex_match_set This is AWS WAF Classic documentation
delete_regex_pattern_set This is AWS WAF Classic documentation
delete_rule This is AWS WAF Classic documentation
delete_rule_group This is AWS WAF Classic documentation
delete_size_constraint_set This is AWS WAF Classic documentation
delete_sql_injection_match_set This is AWS WAF Classic documentation
delete_web_acl This is AWS WAF Classic documentation
delete_xss_match_set This is AWS WAF Classic documentation
disassociate_web_acl This is AWS WAF Classic Regional documentation
get_byte_match_set This is AWS WAF Classic documentation
get_change_token This is AWS WAF Classic documentation
get_change_token_status This is AWS WAF Classic documentation
get_geo_match_set This is AWS WAF Classic documentation
get_ip_set This is AWS WAF Classic documentation
get_logging_configuration This is AWS WAF Classic documentation
get_permission_policy This is AWS WAF Classic documentation
get_rate_based_rule This is AWS WAF Classic documentation
get_rate_based_rule_managed_keys This is AWS WAF Classic documentation
get_regex_match_set This is AWS WAF Classic documentation
get_regex_pattern_set This is AWS WAF Classic documentation
Examples

```r
## Not run:
svc <- wafregional()

# The following example creates an IP match set named MyIPSetFriendlyName.
```
### Description

**WAF**

This is the latest version of the WAF API, released in November, 2019. The names of the entities that you use to access this API, like endpoints and namespaces, all have the versioning information added, like "V2" or "v2", to distinguish from the prior version. We recommend migrating your resources to this version, because it has a number of significant improvements.

If you used WAF prior to this release, you can’t use this WAFV2 API to access any WAF resources that you created before. You can access your old rules, web ACLs, and other WAF resources only through the WAF Classic APIs. The WAF Classic APIs have retained the prior names, endpoints, and namespaces.

For information, including how to migrate your WAF resources to this version, see the WAF Developer Guide.

WAF is a web application firewall that lets you monitor the HTTP and HTTPS requests that are forwarded to an Amazon CloudFront distribution, Amazon API Gateway REST API, Application Load Balancer, AppSync GraphQL API, Amazon Cognito user pool, App Runner service, or Amazon Web Services Verified Access instance. WAF also lets you control access to your content, to protect the Amazon Web Services resource that WAF is monitoring. Based on conditions that you specify, such as the IP addresses that requests originate from or the values of query strings, the protected resource responds to requests with either the requested content, an HTTP 403 status code (Forbidden), or with a custom response.

This API guide is for developers who need detailed information about WAF API actions, data types, and errors. For detailed information about WAF features and guidance for configuring and using WAF, see the WAF Developer Guide.

You can make calls using the endpoints listed in WAF endpoints and quotas.

- For regional applications, you can use any of the endpoints in the list. A regional application can be an Application Load Balancer (ALB), an Amazon API Gateway REST API, an App-Sync GraphQL API, an Amazon Cognito user pool, an App Runner service, or an Amazon Web Services Verified Access instance.
- For Amazon CloudFront applications, you must use the API endpoint listed for US East (N. Virginia): us-east-1.
Alternatively, you can use one of the Amazon Web Services SDKs to access an API that's tailored to the programming language or platform that you're using. For more information, see Amazon Web Services SDKs.

We currently provide two versions of the WAF API: this API and the prior versions, the classic WAF APIs. This new API provides the same functionality as the older versions, with the following major improvements:

- You use one API for both global and regional applications. Where you need to distinguish the scope, you specify a Scope parameter and set it to CLOUDFRONT or REGIONAL.
- You can define a web ACL or rule group with a single call, and update it with a single call. You define all rule specifications in JSON format, and pass them to your rule group or web ACL calls.
- The limits WAF places on the use of rules more closely reflects the cost of running each type of rule. Rule groups include capacity settings, so you know the maximum cost of a rule group when you use it.

Usage

```python
wafv2(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **credentials**: Set anonymous credentials.
    - **endpoint**: The complete URL to use for the constructed client.
    - **region**: The AWS Region used in instantiating the client.
    - **close_connection**: Immediately close all HTTP connections.
    - **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
    - **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
    - **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - **creds**:
    - **access_key_id**: AWS access key ID
    - **secret_access_key**: AWS secret access key
– **session_token**: AWS temporary session token
   - **profile**: The name of a profile to use. If not given, then the default profile is used.
   - **anonymous**: Set anonymous credentials.

**endpoint**  
Optional shorthand for complete URL to use for the constructed client.

**region**  
Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- wafv2(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

**Operations**
associate_web_acl
check_capacity
create_api_key
create_ip_set
create_regex_pattern_set
create_rule_group
create_web_acl
delete_api_key
delete_firewall_manager_rule_groups
delete_ip_set
delete_logging_configuration
delete_permission_policy
delete_regex_pattern_set
delete_rule_group
delete_web_acl
describe_all_managed_products
describeManaged_products_by_vendor
describe_managed_rule_group
disassociate_web_acl
generate_mobile_sdk_release_url
get_decrypted_api_key
generate_ip_set
get_logging_configuration
get_managed_rule_set
generate_mobile_sdk_release
get_permission_policy
get_rate_based_statement_managed_keys
get_regex_pattern_set
generate_ip_set
get_sampled_requests
get_web_acl
get_web_acl_for_resource
get_api_keys
get_available_managed_rule_groups
get_available_managed_rule_group_versions
get_ip_sets
get_logging_configurations
get_managed_rule_sets
get_mobile_sdk_releases
get_regex_pattern_sets
get_resources_for_web_acl
get_rule_groups
get_tags_for_resource
get_web_ac_l
put_logging_configuration
put_available_managed_rule_group_versions
put_permission_policy
tag_resource

Associates a web ACL with a regional application resource, to protect the resource.
Returns the web ACL capacity unit (WCU) requirements for a specified scope.
Creates an API key that contains a set of token domains.
Creates an IPSet, which you use to identify web requests that originate from specific IP addresses or ranges of IP addresses.
Creates a RegexPatternSet, which you reference in a RegexPatternSetReferenceStatement.
Creates a RuleGroup per the specifications provided.
Creates a WebACL per the specifications provided.
Deletes the specified API key.
Deletes all rule groups that are managed by Firewall Manager for the specified WebACL.
Deletes the specified IPSet.
Deletes the LoggingConfiguration from the specified web ACL.
Permanently deletes an IAM policy from the specified rule group.
Deletes the specified RuleGroup.
Deletes the specified WebACL.
Provides high-level information for the Amazon Web Services Managed Rule Groups.
Provides high-level information for the managed rule groups owned by a specific vendor.
Provides high-level information for a managed rule group, including descriptive metadata.
Disassociates the specified regional application resource from any existing web ACL association.
Generates a presigned download URL for the specified release of the mobile SDK.
Retrieves your API key in decrypted form.
Retrieves the specified IPSet.
Retrieves the LoggingConfiguration for the specified web ACL.
Retrieves the specified managed rule set.
Retrieves information for the specified mobile SDK release, including release notes and tags.
Retrieves the IP addresses that are currently blocked by a rate-based rule instance.
Retrieves the specified RegexPatternSet.
Retrieves the specified RuleGroup.
Retrieves detailed information about a specified number of requests—a sample—from the specified WebACL.
Retrieves the WebACL for the specified resource.
Retrieves a list of the API keys that you’ve defined for the specified scope.
Retrieves an array of managed rule groups that are available for you to use.
Retrieves a list of the available versions for the specified managed rule group.
Retrieves an array of IPSets that you manage.
Retrieves an array of your LoggingConfiguration objects.
Retrieves the managed rule sets that you own.
Retrieves a list of the available releases for the mobile SDK and the specified platform.
Retrieves an array of RegexPatternSetSummary objects for the regex pattern sets that you own.
Retrieves an array of the Amazon Resource Names (ARNs) for the regional application resource.
Retrieves an array of RuleGroupSummary objects for the rule groups that you own.
Retrieves the TagInfoForResource for the specified resource.
Retrieves an array of WebACLSummary objects for the web ACLs that you own.
Enables the specified LoggingConfiguration, to start logging from a web ACL.
Defines the versions of your managed rule set that you are offering to the customers.
Attaches an IAM policy to the specified resource.
Associates tags with the specified Amazon Web Services resource.
untag_resource  Disassociates tags from an Amazon Web Services resource
update_ip_set  Updates the specified IPSet
update_managed_rule_set_version_expiry_date  Updates the expiration information for your managed rule set
update_regex_pattern_set  Updates the specified RegexPatternSet
update_rule_group  Updates the specified RuleGroup
update_web_acl  Updates the specified WebACL

Examples

## Not run:
svc <- wafv2()
svc$associate_web_acl(
  Foo = 123
)

## End(Not run)

---

**Description**

Well-Architected Tool

This is the *Well-Architected Tool API Reference*. The WA Tool API provides programmatic access to the Well-Architected Tool in the Amazon Web Services Management Console. For information about the Well-Architected Tool, see the *Well-Architected Tool User Guide*.

**Usage**

```r
wellarchitected(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

**Arguments**

- `config`: Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
    - `creds`:
      * `access_key_id`: AWS access key ID
      * `secret_access_key`: AWS secret access key
      * `session_token`: AWS temporary session token
• **profile**: The name of a profile to use. If not given, then the default profile is used.
• **anonymous**: Set anonymous credentials.

• **endpoint**: The complete URL to use for the constructed client.
• **region**: The AWS Region used in instantiating the client.
• **close_connection**: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
• **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

**credentials**
Optional credentials shorthand for the config parameter

• **creds**:
  • **access_key_id**: AWS access key ID
  • **secret_access_key**: AWS secret access key
  • **session_token**: AWS temporary session token
• **profile**: The name of a profile to use. If not given, then the default profile is used.
• **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**
A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- wellarchitected(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
  )
)
```
WellArchitected

```python
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)
```

**Operations**

- `associate_lenses` - Associate a lens to a workload
- `associate_profiles` - Associate a profile with a workload
- `create_lens_share` - Create a lens share
- `create_lens_version` - Create a new lens version
- `create_milestone` - Create a milestone for an existing workload
- `create_profile` - Create a profile
- `create_profile_share` - Create a profile share
- `create_review_template` - Create a review template
- `create_template_share` - Create a review template share
- `create_workload` - Create a new workload
- `create_workload_share` - Create a workload share
- `delete_lens` - Delete an existing lens
- `delete_lens_share` - Delete a lens share
- `delete_profile` - Delete a profile
- `delete_profile_share` - Delete a profile share
- `delete_review_template` - Delete a review template
- `delete_template_share` - Delete a review template share
- `delete_workload` - Delete an existing workload
- `delete_workload_share` - Delete a workload share
- `disassociate_lenses` - Disassociate a lens from a workload
- `disassociate_profiles` - Disassociate a profile from a workload
- `export_lens` - Export an existing lens
- `get_answer` - Get the answer to a specific question in a workload review
- `get_consolidated_report` - Get a consolidated report of your workloads
- `get_global_settings` - Global settings for all workloads
- `get_lens` - Get an existing lens
- `get_lens_review` - Get lens review
- `get_lens_review_report` - Get lens review report
- `get_lens_version_difference` - Get lens version differences
get_milestone
get_profile
get_profile_template
get_review_template
get_review_template_answer
get_review_template_lens_review
get_workload
import_lens
list_answers
list_check_details
list_check_summaries
list_lenses
list_lens_review_improvements
list_lens_reviews
list_lens_shares
list_milestones
list_notifications
list_profile_notifications
list_profiles
list_profile_shares
list_review_template_answers
list_review_templates
list_share_invitations
list_tags_for_resource
list_template_shares
list_workloads
list_workload_shares
tag_resource
untag_resource
update_answer
update_global_settings
update_integration
update_lens_review
update_profile
update_review_template
update_review_template_answer
update_review_template_lens_review
update_share_invitation
update_workload
update_workload_share
upgrade_lens_review
upgrade_profile_version
upgrade_review_template_lens_review

Get a milestone for an existing workload
Get profile information
Get profile template
Get review template
Get review template answer
Get a lens review associated with a review template
Get an existing workload
Import a new custom lens or update an existing custom lens
List of answers for a particular workload and lens
List of Trusted Advisor check details by account related to the workload
List of Trusted Advisor checks summarized for all accounts related to the workload
List the available lenses
List the improvements of a particular lens review
List lens reviews for a particular workload
List the lens shares associated with the lens
List all milestones for an existing workload
List lens notifications
List profile notifications
List profiles
List profile shares
List the answers of a review template
List review templates
List the share invitations
List the tags for a resource
List review template shares
Paginated list of workloads
List the workload shares associated with the workload
Adds one or more tags to the specified resource
Deletes specified tags from a resource
Update the answer to a specific question in a workload review
Update whether the Amazon Web Services account is opted into organization sharing
Update integration features
Update lens review for a particular workload
Update a profile
Update a review template
Update a review template answer
Update a lens review associated with a review template
Update a workload or custom lens share invitation
Update an existing workload
Update a workload share
Upgrade lens review for a particular workload
Upgrade a profile
Upgrade the lens review of a review template

Examples

## Not run:
svc <- wellarchitected()
svc$associate_lenses(
  Foo = 123
)
## End(Not run)

---

**Description**

The Amazon WorkDocs API is designed for the following use cases:

- **File Migration**: File migration applications are supported for users who want to migrate their files from an on-premises or off-premises file system or service. Users can insert files into a user directory structure, as well as allow for basic metadata changes, such as modifications to the permissions of files.

- **Security**: Security applications are supported for users who have additional security needs, such as antivirus or data loss prevention. The API actions, along with CloudTrail, allow these applications to detect when changes occur in Amazon WorkDocs. Then, the application can take the necessary actions and replace the target file. If the target file violates the policy, the application can also choose to email the user.

- **eDiscovery/Analytics**: General administrative applications are supported, such as eDiscovery and analytics. These applications can choose to mimic or record the actions in an Amazon WorkDocs site, along with CloudTrail, to replicate data for eDiscovery, backup, or analytical applications.

All Amazon WorkDocs API actions are Amazon authenticated and certificate-signed. They not only require the use of the Amazon Web Services SDK, but also allow for the exclusive use of IAM users and roles to help facilitate access, trust, and permission policies. By creating a role and allowing an IAM user to access the Amazon WorkDocs site, the IAM user gains full administrative visibility into the entire Amazon WorkDocs site (or as set in the IAM policy). This includes, but is not limited to, the ability to modify file permissions and upload any file to any user. This allows developers to perform the three use cases above, as well as give users the ability to grant access on a selective basis using the IAM model.

The pricing for Amazon WorkDocs APIs varies depending on the API call type for these actions:

- **READ** (Get*)
- **WRITE** (Activate*, Add*, Create*, Deactivate*, Initiate*, Update*)
- **LIST** (Describe*)
- **DELETE*, CANCEL**

For information about Amazon WorkDocs API pricing, see [Amazon WorkDocs Pricing](#).

**Usage**

```r
workdocs(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```
Arguments

config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
    • profile: The name of a profile to use. If not given, then the default profile is used.
    • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```
svc <- workdocs(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string",
        profile = "string",
        anonymous = FALSE
      )
    )
  )
)
```
secret_access_key = "string",
session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string",
close_connection = "logical",
timeout = "numeric",
s3_force_path_style = "logical",
sts_regional_endpoint = "string"
),
credentials = list(
creds = list(
   access_key_id = "string",
   secret_access_key = "string",
   session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

abort_document_version_upload: Aborts the upload of the specified document version that was previously initiated by InitiateDocumentVersionUpload.
activate_user: Activates the specified user.
add_resource_permissions: Creates a set of permissions for the specified folder or document.
create_comment: Adds a new comment to the specified document version.
create_custom_metadata: Adds one or more custom properties to the specified resource (a folder, document, or version).
create_folder: Creates a folder with the specified name and parent folder.
create_labels: Adds the specified list of labels to the given resource (a document or folder).
create_notification_subscription: Configure Amazon WorkDocs to use Amazon SNS notifications.
create_user: Creates a user in a Simple AD or Microsoft AD directory.
deactivate_user: Deactivates the specified user, which revokes the user's access to Amazon WorkDocs.
delete_comment: Deletes the specified comment from the document version.
delete_custom_metadata: Deletes custom metadata from the specified resource.
delete_document: Permanently deletes the specified document and its associated metadata.
delete_document_version: Deletes a specific version of a document.
delete_folder: Permanently deletes the specified folder and its contents.
delete_folder_contents: Deletes the contents of the specified folder.
delete_labels: Deletes the specified list of labels from a resource.
delete_notification_subscription: Deletes the specified subscription from the specified organization.
delete_user: Deletes the specified user from a Simple AD or Microsoft AD directory.
describe_activities: Describes the user activities in a specified time period.
describe_comments
describe_document_versions
describe_folder_contents
describe_groups
describe_notification_subscriptions
describe_resource_permissions
describe_root_folders
describe_users
get_current_user
get_document
get_document_path
get_document_version
get_folder
get_folder_path
get_resources
initiate_document_version_upload
remove_all_resource_permissions
remove_resource_permission
restore_document_versions
search_resources
update_document
update_document_version
update_folder
update_user

List all the comments for the specified document version
Retrieves the document versions for the specified document
Describes the contents of the specified folder, including its documents and subfolders
Describes the groups specified by the query
Lists the specified notification subscriptions
Describes the permissions of a specified resource
Describes the current user's special folders; the RootFolder and the RecycleBin
Describes the specified users
Retrieves details of the current user for whom the authentication token was generated
Retrieves details of a document
Retrieves the path information (the hierarchy from the root folder) for the requested document
Retrieves version metadata for the specified document
Retrieves the metadata of the specified folder
Retrieves the path information (the hierarchy from the root folder) for the specified folder
Retrieves a collection of resources, including folders and documents
Creates a new document object and version object
Removes all the permissions from the specified resource
Removes the permission for the specified principal from the specified resource
Recovers a deleted version of an Amazon WorkDocs document
Searches metadata and the content of folders, documents, document versions, and comments
Updates the specified attributes of a document
Changes the status of the document version to ACTIVE
Updates the specified attributes of the specified folder
Updates the specified attributes of the specified user, and grants or revokes administrative

Examples

```r
## Not run:
svc <- workdocs()
svc$abort_document_version_upload(
  Foo = 123
)

## End(Not run)
```

Amazon WorkLink

Amazon WorkLink is a cloud-based service that provides secure access to internal websites and web apps from iOS and Android phones. In a single step, your users, such as employees, can access internal websites as efficiently as they access any other public website. They enter a URL in their web browser, or choose a link to an internal website in an email. Amazon WorkLink authenticates
worklink

the user’s access and securely renders authorized internal web content in a secure rendering service in the AWS cloud. Amazon WorkLink doesn’t download or store any internal web content on mobile devices.

Usage

worklink(config = list(), credentials = list(), endpoint = NULL, region = NULL)

Arguments

config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ent.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

default Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```
svc <- worklink(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- **associate_domain**
  - Specifies a domain to be associated to Amazon WorkLink
- **associate_website_authorization_provider**
  - Associates a website authorization provider with a specified fleet
- **associate_website_certificate_authority**
  - Imports the root certificate of a certificate authority (CA) used to obtain TLS certificates
- **create_fleet**
  - Creates a fleet
- **delete_fleet**
  - Deletes a fleet
- **describe_audit_stream_configuration**
  - Describes the configuration for delivering audit streams to the customer account
- **describe_company_network_configuration**
  - Describes the networking configuration to access the internal websites associated with the specified fleet
- **describe_device**
  - Provides information about a user’s device
- **describe_device_policy_configuration**
  - Describes the device policy configuration for the specified fleet
- **describe_domain**
  - Provides information about the domain
- **describe_fleet_metadata**
  - Provides basic information for the specified fleet, excluding identity provider, networking, and device configuration details
- **describe_identity_provider_configuration**
  - Describes the identity provider configuration of the specified fleet
- **describe_website_certificate_authority**
  - Provides information about the certificate authority
- **disassociate_domain**
  - Disassociates a domain from Amazon WorkLink
disassociate_website_authorization_provider  Disassociates a website authorization provider from a specified fleet
disassociate_website_certificate_authority  Removes a certificate authority (CA)
list_devices  Retrieves a list of devices registered with the specified fleet
list_domains  Retrieves a list of domains associated to a specified fleet
list_fleets  Retrieves a list of fleets for the current account and Region
list_tags_for_resource  Retrieves a list of tags for the specified resource
list_website_authorization_providers  Retrieves a list of website authorization providers associated with a specified fleet
list_website_certificateAuthorities  Retrieves a list of certificate authorities added for the current account and Region
restore_domain_access  Moves a domain to ACTIVE status if it was in the INACTIVE status
revoke_domain_access  Moves a domain to INACTIVE status if it was in the ACTIVE status
sign_out_user  Signs the user out from all of their devices
tag_resource  Adds or overwrites one or more tags for the specified resource, such as a fleet
untag_resource  Removes one or more tags from the specified resource
update_audit_stream_configuration  Updates the audit stream configuration for the fleet
update_company_network_configuration  Updates the company network configuration for the fleet
update_device_policy_configuration  Updates the device policy configuration for the fleet
update_domain_metadata  Updates domain metadata, such as DisplayName
update_fleet_metadata  Updates fleet metadata, such as DisplayName
update_identity_provider_configuration  Updates the identity provider configuration for the fleet

Examples

```r
## Not run:
svc <- worklink()
svc$associate_domain(
  Foo = 123
)

## End(Not run)
```

---

**workmail**  
*Amazon WorkMail*

**Description**

WorkMail is a secure, managed business email and calendaring service with support for existing desktop and mobile email clients. You can access your email, contacts, and calendars using Microsoft Outlook, your browser, or other native iOS and Android email applications. You can integrate WorkMail with your existing corporate directory and control both the keys that encrypt your data and the location in which your data is stored.

The WorkMail API is designed for the following scenarios:

- Listing and describing organizations
- Managing users
• Managing groups
• Managing resources

All WorkMail API operations are Amazon-authenticated and certificate-signed. They not only require the use of the AWS SDK, but also allow for the exclusive use of AWS Identity and Access Management users and roles to help facilitate access, trust, and permission policies. By creating a role and allowing an IAM user to access the WorkMail site, the IAM user gains full administrative visibility into the entire WorkMail organization (or as set in the IAM policy). This includes, but is not limited to, the ability to create, update, and delete users, groups, and resources. This allows developers to perform the scenarios listed above, as well as give users the ability to grant access on a selective basis using the IAM model.

Usage

workmail(config = list(), credentials = list(), endpoint = NULL, region = NULL)

Arguments

config Optional configuration of credentials, endpoint, and/or region.
  • credentials:
    – creds:
      * access_key_id: AWS access key ID
      * secret_access_key: AWS secret access key
      * session_token: AWS temporary session token
    – profile: The name of a profile to use. If not given, then the default profile is used.
    – anonymous: Set anonymous credentials.
  • endpoint: The complete URL to use for the constructed client.
  • region: The AWS Region used in instantiating the client.
  • close_connection: Immediately close all HTTP connections.
  • timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
  • s3_force_path_style: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
  • sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest//guide/feature-sts-regionalized-endpoints.html

credentials Optional credentials shorthand for the config parameter
  • creds:
    – access_key_id: AWS access key ID
    – secret_access_key: AWS secret access key
    – session_token: AWS temporary session token
  • profile: The name of a profile to use. If not given, then the default profile is used.
  • anonymous: Set anonymous credentials.

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.
Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- workmail(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  endpoint = "string",
  region = "string"
)
```

Operations

- `associate_delegate_to_resource`: Adds a member (user or group) to the resource’s set of delegates
- `associate_member_to_group`: Adds a member (user or group) to the group’s set
- `assume_impersonation_role`: Assumes an impersonation role for the given WorkMail organization
- `cancel_mailbox_export_job`: Cancels a mailbox export job
- `create_alias`: Adds an alias to the set of a given member (user or group) of WorkMail
- `create_availability_configuration`: Creates an AvailabilityConfiguration for the given WorkMail organization and domain
- `create_group`: Creates a group that can be used in WorkMail by calling the RegisterToWorkMail operation
- `create_impersonation_role`: Creates an impersonation role for the given WorkMail organization
create_mobile_device_access_rule
create_organization
create_resource
create_user
delete_access_control_rule
delete_alias
delete_availability_configuration
delete_email_monitoring_configuration
delete_group
delete_impersonation_role
delete_mailbox_permissions
delete_mobile_device_access_override
delete_mobile_device_access_rule
delete_organization
delete_resource
delete_retention_policy
delete_user
deregister_from_work_mail
deregister_mail_domain
describe_email_monitoring_configuration
describe_entity
describe_group
describe_inbound_dmarc_settings
describe_mailbox_export_job
describe_organization
describe_resource
describe_user
disassociate_delegate_from_resource
disassociate_member_from_group
get_access_control_effect
get_default_retention_policy
get_impersonation_role
get_impersonation_role_effect
get_mailbox_details
get_mail_domain
get_mobile_device_access_effect
get_mobile_device_access_override
list_access_control_rules
list_aliases
list_availability_configurations
list_group_members
list_groups
list_groups_for_entity
list_impersonation_roles
list_mailbox_export_jobs
list_mailbox_permissions
list_mail_domains
list_mobile_device_access_overrides

Creates a new mobile device access rule for the specified WorkMail organization
Creates a new WorkMail organization
Creates a new WorkMail resource
Creates a user who can be used in WorkMail by calling the RegisterToWorkMail operation
Deletes an access control rule for the specified WorkMail organization
Remove one or more specified aliases from a set of aliases for a given user
Deletes the AvailabilityConfiguration for the given WorkMail organization and domain
Deletes the email monitoring configuration for a specified organization
Deletes a group from WorkMail
Deletes an impersonation role for the given WorkMail organization
Deletes permissions granted to a member (user or group)
Deletes the mobile device access override for the given WorkMail organization, user, and device
Deletes a mobile device access rule for the specified WorkMail organization
Deletes an WorkMail organization and all underlying AWS resources managed by WorkMail
Deletes the specified resource
Deletes the specified retention policy from the specified organization
Deletes a user from WorkMail and all subsequent systems
Mark a user, group, or resource as no longer used in WorkMail
Removes a domain from WorkMail, stops email routing to WorkMail, and removes the domain from your AWS organization
Describes the current email monitoring configuration for a specified organization
Returns basic details about an entity in WorkMail
Returns the data available for the group
Lists the settings in a DMARC policy for a specified organization
Describes the current status of a mailbox export job
Provides more information regarding a given organization based on its identifier
Returns the data available for the resource
Provides information regarding the user
Removes a member from the resource’s set of delegates
Removes a member from a group
Gets the effects of an organization’s access control rules as they apply to a specific user or group
Gets the default retention policy details for the specified organization
Gets the impersonation role details for the given WorkMail organization
Tests whether the given impersonation role can impersonate a target user
Requests a user’s mailbox details for a specified organization and user
Gets details for a mail domain, including domain records required to configure your domains
Simulates the effect of the mobile device access rules for the given attributes of a sample access event
Gets the mobile device access override for the given WorkMail organization, user, group, and device
Lists the access control rules for the specified organization
Creates a paginated call to list the aliases associated with a given entity
List all the AvailabilityConfiguration’s for the given WorkMail organization
Returns an overview of the members of a group
Returns summaries of the organization’s groups
Returns all the groups to which an entity belongs
Lists all the impersonation roles for the given WorkMail organization
Lists the mailbox export jobs started for the specified organization within the last 24 hours
Lists the mailbox permissions associated with a user, group, or resource mailbox
Lists the mail domains in a given WorkMail organization
Lists all the mobile device access overrides for any given combination of WorkMail organizations and devices
workmailmessageflow

list_mobile_device_access_rules
list_organizations
list_resource_delegates
list_resources
list_tags_for_resource
list_users
put_access_control_rule
put_email_monitoring_configuration
put_inbound_dmarc_settings
put_mailbox_permissions
put_mobile_device_access_override
put_retention_policy
register_mail_domain
register_to_work_mail
reset_password
start_mailbox_export_job
tag_resource
test_availability_configuration
untag_resource
update_availability_configuration
update_default_mail_domain
update_group
update_impersonation_role
update_mailbox_quota
update_mobile_device_access_rule
update_primary_email_address
update_resource
update_user

Lists the mobile device access rules for the specified WorkMail organization
Returns summaries of the customer’s organizations
Lists the delegates associated with a resource
Returns summaries of the organization’s resources
Lists the tags applied to an WorkMail organization resource
Returns summaries of the organization’s users
Adds a new access control rule for the specified organization
Creates or updates the email monitoring configuration for a specified organization
Enables or disables a DMARC policy for a given organization
Sets permissions for a user, group, or resource
Creates or updates a mobile device access override for the given WorkMail organization
Puts a retention policy to the specified organization
Registers a new domain in WorkMail and SES, and configures it for use by WorkMail
Registers an existing and disabled user, group, or resource for WorkMail use by an administrator
Allows the administrator to reset the password for a user
Starts a mailbox export job to export MIME-format email messages and calendar items
Applies the specified tags to the specified WorkMail organization resource
Performs a test on an availability provider to ensure that access is allowed
Untags the specified tags from the specified WorkMail organization resource
Updates an existing AvailabilityConfiguration for the given WorkMail organization
Updates the default mail domain for an organization
Updates attributes in a group
Updates an impersonation role for the given WorkMail organization
Updates a user’s current mailbox quota for a specified organization and user
Updates a mobile device access rule for the specified WorkMail organization
Updates the primary email for a user, group, or resource
Updates data for the resource
Updates data for the user

Examples

```
## Not run:
svc <- workmail()
svc$associate_delegate_to_resource(
    Foo = 123
)

## End(Not run)
```

workmailmessageflow  Amazon WorkMail Message Flow

Description

The WorkMail Message Flow API provides access to email messages as they are being sent and received by a WorkMail organization.
Usage

```r
workmailmessageflow(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.
  - **creds**:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - `profile`: The name of a profile to use. If not given, then the default profile is used.
  - `anonymous`: Set anonymous credentials.
- **endpoint**: The complete URL to use for the constructed client.
- **region**: The AWS Region used in instantiating the client.
- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html)

- **credentials**: Optional credentials shorthand for the config parameter
  - `creds`:
    - `access_key_id`: AWS access key ID
    - `secret_access_key`: AWS secret access key
    - `session_token`: AWS temporary session token
  - `profile`: The name of a profile to use. If not given, then the default profile is used.
  - `anonymous`: Set anonymous credentials.

- **endpoint**: Optional shorthand for complete URL to use for the constructed client.
- **region**: Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.
Service syntax

```
svc <- `workmailmessageflow`(config = list(
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  ),
  profile = "string",
  anonymous = "logical"
),
  endpoint = "string",
  region = "string",
  close_connection = "logical",
  timeout = "numeric",
  s3_force_path_style = "logical",
  sts_regional_endpoint = "string"
),
credentials = list(
  creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
  ),
  profile = "string",
  anonymous = "logical"
),
  endpoint = "string",
  region = "string"
))
```

Operations

- `get_raw_message_content` Retrieves the raw content of an in-transit email message, in MIME format
- `put_raw_message_content` Updates the raw content of an in-transit email message, in MIME format

Examples

```
## Not run:
svc <- `workmailmessageflow`()
svc$get_raw_message_content(
  Foo = 123
)

## End(Not run)
```
Description

Amazon WorkSpaces Service

Amazon WorkSpaces enables you to provision virtual, cloud-based Microsoft Windows or Amazon Linux desktops for your users, known as WorkSpaces. WorkSpaces eliminates the need to procure and deploy hardware or install complex software. You can quickly add or remove users as your needs change. Users can access their virtual desktops from multiple devices or web browsers.

This API Reference provides detailed information about the actions, data types, parameters, and errors of the WorkSpaces service. For more information about the supported Amazon Web Services Regions, endpoints, and service quotas of the Amazon WorkSpaces service, see WorkSpaces endpoints and quotas in the Amazon Web Services General Reference.

You can also manage your WorkSpaces resources using the WorkSpaces console, Command Line Interface (CLI), and SDKs. For more information about administering WorkSpaces, see the Amazon WorkSpaces Administration Guide. For more information about using the Amazon WorkSpaces client application or web browser to access provisioned WorkSpaces, see the Amazon WorkSpaces User Guide. For more information about using the CLI to manage your WorkSpaces resources, see the WorkSpaces section of the CLI Reference.

Usage

```python
workspaces(
    config = list(),
    credentials = list(),
    endpoint = NULL,
    region = NULL
)
```

Arguments

config Optional configuration of credentials, endpoint, and/or region.

- credentials:
  - creds:
    - access_key_id: AWS access key ID
    - secret_access_key: AWS secret access key
    - session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.
- endpoint: The complete URL to use for the constructed client.
- region: The AWS Region used in instantiating the client.
workspaces

- **close_connection**: Immediately close all HTTP connections.
- **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- **s3_force_path_style**: Set this to `true` to force the request to use path-style addressing, i.e. `http://s3.amazonaws.com/BUCKET/KEY`.
- **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy [https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html](https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoints.html)

**credentials**

Optional credentials shorthand for the config parameter

- **creds**:
  - **access_key_id**: AWS access key ID
  - **secret_access_key**: AWS secret access key
  - **session_token**: AWS temporary session token
- **profile**: The name of a profile to use. If not given, then the default profile is used.
- **anonymous**: Set anonymous credentials.

**endpoint**

Optional shorthand for complete URL to use for the constructed client.

**region**

Optional shorthand for AWS Region used in instantiating the client.

**Value**

A client for the service. You can call the service’s operations using syntax like `svc$operation(...)`, where `svc` is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- workspaces(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
```
creds = list(
    access_key_id = "string",
    secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

accept_account_link_invitation
associate_connection_alias
associate_ip_groups
associate_workspace_application
authorize_ip_rules
copy_workspace_image
create_account_link_invitation
create_connect_client_add_in
create_connection_alias
create_ip_group
create_standby_workspaces
create_tags
create_updated_workspace_image
create_workspace_bundle
create_workspace_image
create_workspaces
delete_account_link_invitation
delete_client_branding
delete_connect_client_add_in
delete_connection_alias
delete_ip_group
delete_tags
delete_workspace_bundle
delete_workspace_image
deploy_workspace_applications
deregister_workspace_directory
describe_account
describe_account_modifications
describe_application_associations
describe_applications
describe_bundle_associations
describe_client_branding
describe_client_properties
describe_connect_client_add_ins

Accepts the account link invitation
Associates the specified connection alias with the specified directory to enable cross-Region redirection
Associates the specified IP access control group with the specified directory
Associates the specified application to the specified WorkSpace
Adds one or more rules to the specified IP access control group
Copies the specified image from the specified Region to the current Region
Creates the account link invitation
Creates a client-add-in for Amazon Connect within a directory
Creates the specified connection alias for use with cross-Region redirection
Creates an IP access control group
Creates a standby WorkSpace in a secondary Region
Creates the specified tags for the specified WorkSpaces resource
Creates a new updated WorkSpace image based on the specified source image
Creates the specified WorkSpace bundle
Creates a new WorkSpace image from an existing WorkSpace
Creates one or more WorkSpaces
Deletes the account link invitation
Deletes customized client branding
Deletes a client-add-in for Amazon Connect that is configured within a directory
Deletes the specified connection alias
Deletes the specified IP access control group
Deletes the specified tags from the specified WorkSpaces resource
Deletes the specified WorkSpace bundle
Deletes the specified image from your account
Deploys associated applications to the specified WorkSpace
Deregisters the specified directory
Retrieves a list that describes the configuration of Bring Your Own License (BYOL)
Retrieves a list that describes modifications to the configuration of Bring Your Own License (BYOL)
Describes the associations between the application and the specified associated resource
Describes the specified applications by filtering based on their compute types, licenses, and operating systems
Describes the associations between the applications and the specified bundle
Describes the specified client branding
Retrieves a list that describes one or more specified Amazon WorkSpaces clients
Retrieves a list of Amazon Connect client add-ins that have been created
describe_connection_aliases
describe_connection_alias_permissions
describe_image_associations
describe_tags
describe_workspace_associations
describe_workspace_bundles
describe_workspace_directories
describe_workspace_image_permissions
describe_workspace_images
describe_workspaces
describe_workspaces_connection_status
describe_workspace_snapshots
disassociate_connection_alias
disassociate_ip_groups
disassociate_workspace_application
get_account_link
import_client_branding
import_workspace_image
list_account_links
list_available_management_cidr_ranges
migrate_workspace
modify_account
modify_certificate_based_auth_properties
modify_client_properties
modify_saml_properties
modify_selfservice_permissions
modify_workspace_access_properties
modify_workspace_creation_properties
modify_workspace_properties
modify_workspace_state
reboot_workspaces
rebuild_workspaces
register_workspace_directory
reject_account_link_invitation
restore_workspace
revoke_ip_rules
start_workspaces
stop_workspaces
terminate_workspaces
update_connect_client_add_in
update_connection_alias_permission
update_rules_of_ip_group
update_workspace_bundle
update_workspace_image_permission

Retrieves a list that describes the connection aliases used for cross-Region redirection
Describes the permissions that the owner of a connection alias has granted to another account
Describes the associations between the applications and the specified image
Describes one or more of your IP access control groups
Describes the specified tags for the specified WorkSpaces resource
Describes the associations between applications and the specified WorkSpace
Retrieves a list that describes the available WorkSpace bundles
Describes the available directories that are registered with Amazon WorkSpaces
Describes the permissions that the owner of an image has granted to other Amazon Web Services accounts
Retrieves a list that describes one or more specified images, if the image identifier is provided
Describes the specified WorkSpaces
Describes the connection status of the specified WorkSpaces
Describes the snapshots for the specified WorkSpace
Disassociates a connection alias from a directory
Disassociates the specified IP access control group from the specified directory
Disassociates the specified application from a WorkSpace
Retrieves account link information
Imports client branding
Imports the specified Windows 10 or 11 Bring Your Own License (BYOL) image
Lists all account links
Retrieves a list of IP address ranges, specified as IPv4 CIDR blocks, that you can use for the network management interface
Migrates a WorkSpace from one operating system or bundle type to another, while retaining the data on the user volume
Modifies the configuration of Bring Your Own License (BYOL) for the specified account
Modifies the properties of the specified Amazon WorkSpaces clients
Modifies multiple properties related to SAML 2
Modifies the self-service WorkSpace management capabilities for your users
Specifies which devices and operating systems users can use to access their WorkSpaces
Modifies the default properties used to create WorkSpaces
Modifies the specified WorkSpace properties
Sets the state of the specified WorkSpace
Reboots the specified WorkSpaces
Rebuilds the specified WorkSpace
Registers the specified directory
Rejects the account link invitation
Restores the specified WorkSpace to its last known healthy state
Removes one or more rules from the specified IP access control group
Starts the specified WorkSpaces
Stops the specified WorkSpaces
Terminates the specified WorkSpaces
Updates a Amazon Connect client add-in
Shares or unshares a connection alias with one account by specifying whether that account has permission to associate the connection alias with a directory
Replaces the current rules of the specified IP access control group with the specified rules
Updates a WorkSpace bundle with a new image
Shares or unshares an image with one account in the same Amazon Web Services Region
Examples

```r
## Not run:
svc <- workspaces()
svc$accept_account_link_invitation(
  Foo = 123
)
## End(Not run)
```

--

workspacesweb  

Amazon WorkSpaces Web

Description

WorkSpaces Web is a low cost, fully managed WorkSpace built specifically to facilitate secure, web-based workloads. WorkSpaces Web makes it easy for customers to safely provide their employees with access to internal websites and SaaS web applications without the administrative burden of appliances or specialized client software. WorkSpaces Web provides simple policy tools tailored for user interactions, while offloading common tasks like capacity management, scaling, and maintaining browser images.

Usage

```r
workspacesweb(
  config = list(),
  credentials = list(),
  endpoint = NULL,
  region = NULL
)
```

Arguments

- `config`: Optional configuration of credentials, endpoint, and/or region.
  - `credentials`:
    - `creds`:
      - `access_key_id`: AWS access key ID
      - `secret_access_key`: AWS secret access key
      - `session_token`: AWS temporary session token
    - `profile`: The name of a profile to use. If not given, then the default profile is used.
  - `anonymous`: Set anonymous credentials.
  - `endpoint`: The complete URL to use for the constructed client.
  - `region`: The AWS Region used in instantiating the client.
  - `close_connection`: Immediately close all HTTP connections.
• **timeout**: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.

• **s3_force_path_style**: Set this to true to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.

• **sts_regional_endpoint**: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-endpoint.html

**credentials**
Optionally credentials shorthand for the config parameter

• **creds**:
  – **access_key_id**: AWS access key ID
  – **secret_access_key**: AWS secret access key
  – **session_token**: AWS temporary session token

• **profile**: The name of a profile to use. If not given, then the default profile is used.

• **anonymous**: Set anonymous credentials.

**endpoint**
Optional shorthand for complete URL to use for the constructed client.

**region**
Optional shorthand for AWS Region used in instantiating the client.

**Value**
A client for the service. You can call the service’s operations using syntax like svc$operation(...), where svc is the name you’ve assigned to the client. The available operations are listed in the Operations section.

**Service syntax**

```r
svc <- workspacesweb(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = "logical"
    ),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
  ),
  credentials = list(
    creds = list(
      access_key_id = "string",
      secret_access_key = "string",
      session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
  )
)
```
secret_access_key = "string",
session_token = "string"
),
profile = "string",
anonymous = "logical"
),
endpoint = "string",
region = "string"
)

Operations

associate_browser_settings
associate_ip_access_settings
associate_network_settings
associate_trust_store
associate_user_access_logging_settings
associate_user_settings
create_browser_settings
create_identity_provider
create_ip_access_settings
create_network_settings
create_portal
create_trust_store
create_user_access_logging_settings
create_user_settings
delete_browser_settings
delete_identity_provider
delete_ip_access_settings
delete_network_settings
delete_portal
delete_trust_store
delete_user_access_logging_settings
delete_user_settings
disable_browser_settings
disable_ip_access_settings
disable_network_settings
disable_trust_store
disable_user_access_logging_settings
disable_user_settings
get_browser_settings
get_identity_provider
get_ip_access_settings
get_network_settings
get_portal
get_portal_service_provider_metadata
get_trust_store
get_trust_store_certificate

Associates a browser settings resource with a web portal
Associates an IP access settings resource with a web portal
Associates a network settings resource with a web portal
Associates a trust store with a web portal
Associates a user access logging settings resource with a web portal
Associates a user settings resource with a web portal
Creates a browser settings resource that can be associated with a web portal
Creates an identity provider resource that is then associated with a web portal
Creates an IP access settings resource that can be associated with a web portal
Creates a network settings resource that can be associated with a web portal
Creates a web portal
Creates a trust store that can be associated with a web portal
Creates a user access logging settings resource that can be associated with a web portal
Deletes browser settings
Deletes the identity provider
Deletes IP access settings
Deletes network settings
Deletes a web portal
Deletes the trust store
Deletes user access logging settings
Deletes user settings
Disassociates browser settings from a web portal
Disassociates IP access settings from a web portal
Disassociates network settings from a web portal
Disassociates a trust store from a web portal
Disassociates user access logging settings from a web portal
Disassociates user settings from a web portal
Gets browser settings
Gets the identity provider
Gets the IP access settings
Gets the network settings
Gets the web portal
Gets the service provider metadata
Gets the trust store
Gets the trust store certificate
### Description
Amazon Web Services X-Ray provides APIs for managing debug traces and retrieving service maps and other data created by processing those traces.

### Usage
```
xray(config = list(), credentials = list(), endpoint = NULL, region = NULL)
```
Arguments

config Optional configuration of credentials, endpoint, and/or region.

- credentials:
  - creds:
    * access_key_id: AWS access key ID
    * secret_access_key: AWS secret access key
    * session_token: AWS temporary session token
  - profile: The name of a profile to use. If not given, then the default profile is used.
  - anonymous: Set anonymous credentials.
- endpoint: The complete URL to use for the constructed client.
- region: The AWS Region used in instantiating the client.
- close_connection: Immediately close all HTTP connections.
- timeout: The time in seconds till a timeout exception is thrown when attempting to make a connection. The default is 60 seconds.
- s3_force_path_style: Set this to True to force the request to use path-style addressing, i.e. http://s3.amazonaws.com/BUCKET/KEY.
- sts_regional_endpoint: Set sts regional endpoint resolver to regional or legacy https://docs.aws.amazon.com/sdkref/latest/guide/feature-sts-regionalized-ends.html

credentials Optional credentials shorthand for the config parameter

- creds:
  - access_key_id: AWS access key ID
  - secret_access_key: AWS secret access key
  - session_token: AWS temporary session token
- profile: The name of a profile to use. If not given, then the default profile is used.
- anonymous: Set anonymous credentials.

documentation

endpoint Optional shorthand for complete URL to use for the constructed client.

region Optional shorthand for AWS Region used in instantiating the client.

Value

A client for the service. You can call the service's operations using syntax like svc$operation(...), where svc is the name you've assigned to the client. The available operations are listed in the Operations section.

Service syntax

```r
svc <- xray(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string",
      anonymous = FALSE
    )
  ),
  endpoint = "string",
  region = "string",
  close_connection = FALSE,
  timeout = 60,
  s3_force_path_style = FALSE,
  sts_regional_endpoint = "string"
)
```
secret_access_key = "string",
    session_token = "string"
),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string",
    close_connection = "logical",
    timeout = "numeric",
    s3_force_path_style = "logical",
    sts_regional_endpoint = "string"
),
credentials = list(
    creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
    ),
    profile = "string",
    anonymous = "logical"
),
    endpoint = "string",
    region = "string"
)

Operations

batch_get_traces Retrieves a list of traces specified by ID
create_group Creates a group resource with a name and a filter expression
create_sampling_rule Creates a rule to control sampling behavior for instrumented applications
delete_group Deletes a group resource
delete_resource_policy Deletes a resource policy from the target Amazon Web Services account
delete_sampling_rule Deletes a sampling rule
get_encryption_config Retrieves the current encryption configuration for X-Ray data
group Retrieves group resource details
groups Retrieves all active group details
get_insight Retrieves the summary information of an insight
get_insight_events X-Ray reevaluates insights periodically until they’re resolved, and records each intermediate state as an event
get_insight_impact_graph Retrieves a service graph structure filtered by the specified insight
group_summaries Retrieves the summaries of all insights in the specified group
get_sampling_rules Retrieves all sampling rules
group_summaries Retrieves information about recent sampling results for all sampling rules
group_stats Retrieves a service graph for one or more specific trace IDs
group_statistic_changes Retrieves a service graph that describes services that process incoming requests, and downstream requests
get_time_series_service_statistics Retrieves IDs and annotations for traces available for a specified time frame using an optional time range
list_resource_policies  Returns the list of resource policies in the target Amazon Web Services account
list_tags_for_resource  Returns a list of tags that are applied to the specified Amazon Web Services X-Ray group or sampling rule
put_encryption_config  Updates the encryption configuration for X-Ray data
put_resource_policy  Sets the resource policy to grant one or more Amazon Web Services services and accounts permissions to access X-Ray
put_telemetry_records  Used by the Amazon Web Services X-Ray daemon to upload telemetry
put_trace_segments  Uploads segment documents to Amazon Web Services X-Ray
tag_resource  Applies tags to an existing Amazon Web Services X-Ray group or sampling rule
untag_resource  Removes tags from an Amazon Web Services X-Ray group or sampling rule
update_group  Updates a group resource
update_sampling_rule  Modifies a sampling rule’s configuration

Examples

```r
## Not run:
svc <- xray()
svc$batch_get_traces(
  Foo = 123
)
## End(Not run)
```
Index

abort_document_version_upload, 929
abort_environment_update, 354
abort_multipart_read_set_upload, 604
abort_multipart_upload, 418, 757
abort_vault_lock, 418
accept_account_link_invitation, 942
accept_address_transfer, 315
acceptAdministrator_invitation, 436, 799
accept_attachment, 597
accept_direct_connect_gateway_association_proposal, 288
accept_domain_transfer_from_another_aws_account, 741
accept_environment_account_connection, 678
accept_eulas, 601
accept_grant, 530
accept_handshake, 626
accept_inbound_connection, 611
accept_inbound_cross_cluster_search_connection, 360
accept_invitation, 281, 436, 557, 799
accept_page, 848
accept_portfolio_share, 810
accept_predictions, 273
acceptqualification_request, 580
accept_reserved_instances_exchange_quote, 316
accept_reserved_node_exchange, 708
accept_resource_share_invitation, 693
accept_share, 604
accept_shared_directory, 291
accept_subscription_request, 273
accept_transit_gateway_multicast_domain_association, 316
accept_transit_gateway_peering_attachment, 316
accept_transit_gateway_vpc_attachment, 316
accept_vpc_endpoint_connections, 316
accept_vpc_peering_connection, 316
accessanalyzer, 8
account, 11
acknowledge_job, 198
acknowledge_third_party_job, 198
acm, 13
acmpca, 16
activate_anomaly_detector, 551
activate_contact_channel, 848
activate_evaluation_form, 236
activate_event_source, 142, 382
activate_gateway, 867
activate_key_signing_key, 737
activate_organizations_access, 113
activate_pipeline, 271
activate_type, 113
activate_user, 929
add_application_cloud_watch_logging_option, 500, 503
add_application_input, 500, 503
add_application_input_processing_configuration, 500, 503
add_application_output, 500, 503
add_application_reference_data_source, 500, 503
add_application_vpc_configuration, 503
add_association, 769
add_attachments_to_set, 874
add_attributes_to_findings, 460
add_cache, 867
add_client_id_to_open_id_connect_provider, 445
add_communication_to_case, 874
add_custom_attributes, 214
add_custom_routing_endpoints, 422
add_data_source, 611
add_draft_app_version_resource_mappings,
add_endpoints, 422
add_facet_to_object, 109
add_instance_fleet, 370
add_instance_groups, 370
add_ip_routes, 291
add_job_flow_steps, 370
add_layer_version_permission, 514
add_lf_tags_to_resource, 510
add_notification_channel, 285
add_notification_channels, 188
add_partner, 708
add_permission, 514, 836, 840
add_policy_statement, 379
add_profile_key, 267
add_region, 291
add_resource_permissions, 929
add_role_to_db_cluster, 586, 697
add_role_to_db_instance, 697
add_role_to_instance_profile, 445
add_source_identifier_to_subscription, 297, 586, 697
add_tags, 133, 271, 360, 363, 366, 370, 554, 611, 769
add_tags_to_certificate, 15
add_tags_to_on_premises_instances, 184
add_tags_to_resource, 122, 291, 297, 350, 586, 697, 843, 867
allow_custom_routing_traffic, 422
allocate_address, 316
allocate_connection_on_interconnect, 288
allocate_hosted_connection, 288
allocate_hosts, 316
allocate_ipam_pool_cidr, 316
allocate_private_virtual_interface, 288
allocate_public_virtual_interface, 288
allocate_static_ip, 539
allocate_transit_virtual_interface, 288
apply_archive_rule, 10
apply_environment_managed_action, 354
apply_pending_maintenance_action, 297, 586, 697
apply_schema, 109
apply_security_groups_to_client_vpn_target_network, 316
apply_security_groups_to_load_balancer, 363
admin_link_provider_for_user, 215
admin_list_devices, 215
admin_list_groups_for_user, 215
admin_list_user_auth_events, 215
admin_remove_user_from_group, 215
admin_reset_user_password, 215
admin_respond_to_auth_challenge, 215
admin_set_user_mfa_preference, 215
admin_set_user_password, 215
admin_set_user_settings, 215
admin_update_auth_event_feedback, 215
admin_update_device_status, 215
admin_update_user_attributes, 215
admin_user_global_sign_out, 215
advertise_byoip_cidr, 316, 422
analyze_document, 889
analyze_expense, 889
analyze_id, 889
apigateway, 19
apigatewaymanagementapi, 23
apigatewayv2, 26
appfabric, 30
applicationautoscaling, 32
applicationcostprofiler, 36
applicationinsights, 38
apply_archive_rule, 10
apply_security_groups_to_client_vpn_target_network, 316
apply_security_groups_to_load_balancer, 363
appmesh, 41
appregistry, 44
approve_assignment, 580
apprunner, 47
appstream, 51
archive_findings, 436
arczonalshift, 55
assign_instance, 618
assign_ipv_6_addresses, 316
assign_private_ip_addresses, 316
assign_private_nat_gateway_address, 316
assign_tape_pool, 867
assign_volume, 618
associate_access_grants_identity_center, 761
associate_access_policy, 346
associate_accounts, 95
associate_address, 316
associate_admin_account, 402
associate_alias, 117
associate_analytics_data_set, 236
associate_app_block_builder_app_block, 53
associate_application_fleet, 53
associate_application_to_entitlement, 53
associate_approval_rule_template_with_repository, 179
associate_approved_origin, 236
associate_assessment_report_evidence_folder, 65
associate_attribute_group, 46
associate_bot, 236
associate_browser_settings, 946
associate_budget_with_resource, 810
associate_client_vpn_target_network, 316
associate_connect_peer, 597
associate_connection_alias, 942
associate_connection_with_lag, 288
associate_custom_domain, 49
associate_customer_gateway, 597
associate_data_share_consumer, 708
associate_default_view, 729
associate_default_vocabulary, 236
associate_delegation_to_resource, 935
associate_delegation_signer_to_domain, 741
associate_dhcp_options, 316
associate_domain, 932
associate_drt_log_bucket, 831
associate_drt_rule, 831
associate_elastic_ip, 618
associate_enclave_certificate_iam_role, 316
associate_encryption_config, 346
associate_entities_to_experience, 488
associate_environment_operations_role, 354
associate_external_connection, 164
associate_faces, 721
associate_file_system, 867
associate_file_system_aliases, 415
associate_firewall_policy, 594
associate_firewall_rule_group, 753
associate_fleet, 53
associate_flow, 236
associate_fraudster, 906
associate_gateway_to_server, 81
associate_health_check, 831
associate_hosted_connection, 288
associate_iam_instance_profile, 316
associate_identity_provider_config, 346
associate_instance_event_window, 316
associate_instance_storage_config, 236
associate_ip_access_settings, 946
associate_ip_groups, 942
associate_ipam_byoasn, 316
associate_ipam_resource_discovery, 316
associate_kms_key, 152
associate_lambda_function, 236
associate_lenses, 925
associate_lex_bot, 236
associate_license, 561
associate_link, 597
associate_mac_sec_key, 288
associate_member, 463
associate_member_to_group, 935
associate_nat_gateway_address, 316
associate_network_settings, 946
associate_node, 623
associate_ops_item_related_item, 843
associate_origination_identity, 664
associate_package, 360, 611
associate_personas_to_entities, 488
associate_phone_number_contact_flow, 236
associate_pricing_rules, 95
associate_principal_with_portfolio, 810
associate_proactive_engagement_details, 831
associate_product_with_portfolio, 810
associate_profiles, 925
associate_qualification_with_worker, 580
associate_queue_quick_connects, 236
associate_repository, 191
associate_resolver_endpoint_ip_address, 753
associate_resolver_query_log_config, 753
associate_resolve_rule, 753
associate_resource, 46, 883
associate_resource_share, 693
associate_resource_share_permission, 693
associate_route_table, 316
associate_routing_profile_queues, 236
associate_security_key, 236
associate_service_action_with_provisioning_artifact, 810
associate_service_quota_template, 817
associate_software_token, 215
associate_source_network_stack, 303
associate_subnet_cidr_block, 316
associate_subnets, 594
associate_tag_option_with_resource, 810
associate_team_member, 201
associate_third_party_firewall, 402
associate_tracker_consumer, 544
associate_traffic_distribution_group_user, 236
associate_transit_gateway_connect_peer, 597
associate_transit_gateway_multicast_domain, 316
associate_transit_gateway_policy_table, 316
associate_transit_gateway_route_table, 316
associate_trial_component, 769
associate_trunk_interface, 316
associate_trust_store, 946
associate_user, 536
associate_user_access_logging_settings, 946
associate_user_proficiencies, 237
associate_user_settings, 946
associate_user_to_permission_group, 394
associate_virtual_interface, 288
associate_vpc_cidr_block, 316
associate_vpc_with_hosted_zone, 737
associate_web_acl, 917, 922
associate_website_authorization_provider, 932
associate_website_certificate_authority, 932
associate_workspace_application, 942
assume_decorated_role_with_saml, 510
assume_impersonation_role, 935
assume_role, 871
assume_role_with_saml, 871
assume_role_with_web_identity, 871
athena, 58
attach_certificate_to_distribution, 539
attach_classic_link_vpc, 316
attach_customer_managed_policy_reference_to_permission_set, 859
attach_disk, 539
attach_elastic_load_balancer, 618
attach_group_policy, 445
attach_instances, 71
attach_instances_to_load_balancer, 539
attach_internet_gateway, 316
attach_load_balancer_target_groups, 71
attach_load_balancer_tls_certificate, 539
attach_load_balancer_to_subnets, 363
attach_load_balancers, 71
attach_managed_policy_to_permission_set, 859
attach_network_interface, 316
attach_object, 109
attach_policy, 109, 626
attach_role_policy, 445
attach_static_ip, 539
attach_to_index, 109
attach_traffic_sources, 71
attach_typed_link, 109
attach_user_policy, 445
attach_verified_access_trust_provider, 316
attach_volume, 316, 867
attach_vpn_gateway, 316
auditmanager, 61
augmentedairuntime, 66
authorize_cache_security_group_ingress, 350
authorize_client_vpn_ingress, 316
authorize_cluster_security_group_ingress, 708
authorize_data_share, 708
authorize_db_security_group_ingress, 697
authorize_endpoint_access, 708
authorize_ip_rules, 942
authorize_security_group_egress, 316
authorize_security_group_ingress, 316
authorize_snapshot_access, 708
authorize_vpc_endpoint_access, 360, 611
autoscaling, 69
autoscalingplans, 73
back_test_anomaly_detector, 551
backtrack_db_cluster, 697
backup, 75
backupgateway, 79
backupstorage, 82
batch, 85
batch_apply_update_action, 350
batch_associate_analytics_data_set, 237
batch_associate_approval_rule_template_with_repositories, 179
batch_associate_assessment_report_evidence, 65
batch_associate_resource, 402
batch_associate_resources_to_custom_line_item, 95
batch_associate_scram_secret, 483
batch_associate_service_action_with_provisioning_artifact, 810
batch_associate_user_stack, 53
batch_check_layer_availability, 333, 337
batch_check_layer_availability, 333, 337
batch_create_delegation_by_assessment, 65
batch_create_partition, 426
batch_create_rum_metric_definitions, 159
batch_create_variable, 411
batch_delete_attributes, 834
batch_delete_automation_rules, 799
batch_delete_builds, 168
batch_delete_cluster_snapshots, 708
batch_delete_connection, 426
batch_delete_custom_vocabulary_item, 521
batch_delete_delegation_by_assessment, 65
batch_delete_device_position_history, 544
batch_delete_document, 488
batch_delete_featured_results_set, 488
batch_delete_geofoence, 544
batch_delete_image, 333, 337
batch_delete_partition, 426
batch_delete_read_set, 604
batch_delete_recipe_version, 432
batch_delete_rum_metric_definitions, 159
batch_delete_scheduled_action, 71
batch_delete_table, 426
batch_delete_table_version, 426
batch_delete_unique_id, 379
batch_describe_entities, 565
batch_describe_merge_conflicts, 179
batch_describe_model_package, 769
batch_describe_type_configurations, 113
batch_detect_dominant_language, 222
batch_detect_entities, 222
batch_detect_key_phrases, 222
batch_detect_sentiment, 222
batch_detect_syntax, 222
batch_detect_targeted_sentiment, 222
batch_disassociate_analytics_data_set, 237
batch_disassociate_approval_rule_template_from_repositories, 179
batch_disassociate_automation_rule, 799
batch_disassociate_repositories, 179
batch_disassociate_service_action_artifacts, 799
batch_disassociate_users, 53
batch_put_document, 488
batch_put_field_options, 246
batch_put_geofence, 545
batch_put_metrics, 785
batch_put_scheduled_update_group_action, 71
batch_read, 109
batch_revoke_permissions, 510
batch_start_viewer_session_revocation, 470
batch_stop_job_run, 426
batch_stop_update_action, 350
batch_update_automation_rules, 799
batch_update_cluster, 575
batch_update_custom_vocabulary_item, 521
batch_update_device_position, 545
batch_update_findings, 799
batch_update_member_ec_2_deep_inspection_status, 464
batch_update_partition, 426
batch_update_recommendation_status, 725
batch_update_rule, 909
batch_update_standards_control_associations, 799
batch_write, 109
bedrock, 88
bedrockruntime, 90
begin_transaction, 703
billingconductor, 93
braket, 96
budgets, 98
build_auth_token, 697
build_bot_locale, 521
build_suggesters, 127
bulk_publish, 219
bundle_instance, 316
calculate_route, 545
calculate_route_matrix, 545
cancel_annnotation_import_job, 604
cancel_archival, 867
cancel_batch_import_job, 411
cancel_batch_prediction_job, 411
cancel_bundle_task, 316
cancel_capacity_reservation, 60, 316
cancel_capacity_reservation_fleets, 316
cancel_change_set, 565
cancel_command, 843
cancel_component_deployment, 678
cancel_conversion_task, 316
cancel_data_quality_rule_recommendation_run, 426
cancel_data_quality_ruleset_evaluation_run, 426
cancel_data_repository_task, 415
cancel_domain_config_change, 360, 611
cancel_domain_transfer_to_another_aws_account, 741
cancel_elasticsearch_service_software_update, 360
cancel_environment_deployment, 678
cancel_export_job, 823
cancel_environment_deployment, 678
cancel_findings_report, 464
cancel_gremlin_query, 590
cancel_handshake, 626
cancel_image_creation, 457
cancel_image_launch_permission, 316
cancel_import_task, 316
cancel_ingestion, 688
cancel_instance_refresh, 71
cancel_job, 87, 98
cancel_job_run, 373, 376
cancel_journal_kinesis_stream, 681
cancel_key_deletion, 507
cancel_legal_hold, 77
cancel_lifecycle_execution, 457
cancel_loader_job, 590
cancel_mailbox_export_job, 935
cancel_maintenance_window_execution, 843
cancel_message_move_task, 840
cancel_metadata_generation_run, 273
cancel_ml_data_processing_job, 590
cancel_ml_model_training_job, 590
cancel_ml_model_transform_job, 590
cancel_ml_task_run, 426
cancel_open_cypher_query, 590
cancel_policy_generation, 10
cancel_quantum_task, 98
cancel_query, 133, 891
cancel_replay, 142, 382
cancel_reserved_instances_listing, 316
cancel_resize, 708
cancel_resource_request, 106
cancel_retrieval, 867
cancel_rotate_secret, 795
cancel_run, 604
cancel_sbom_export, 464
cancel_schema_extension, 291
cancel_service_instance_deployment, 678
cancel_service_pipeline_deployment, 678
cancel_service_software_update, 611
cancel_sol_network_operation, 886
cancel_spot_fleet_requests, 317
cancel_spot_instance_requests, 317
cancel_statement, 426, 713
cancel_steps, 370
cancel_subscription, 273
cancel_transaction, 510
cancel_update_stack, 113
cancel_variant_import_job, 604
cancel_zonal_shift, 57
change_cidr_collection, 737
change_message_visibility, 840
change_message_visibility_batch, 840
change_password, 215, 445
change_resource_record_sets, 737
change_tags_for_resource, 737
check_access_not_granted, 10
check_capacity, 922
check_dns_availability, 354
check_domain_availability, 741
check_domain_transferability, 741
check_if_phone_number_is_opted_out, 836
check_in_license, 531
check_no_new_access, 10
check_schema_version_validity, 426
checkout_borrow_license, 531
checkout_license, 531
claim_phone_number, 237
classify_document, 222
clear_query_suggestions, 488
close_account, 627
close_instance_public_ports, 539
cloud9, 101
cloudcontrolapi, 105
clouddirectory, 107
cloudformation, 111
cloudfront, 115
cloudhsm, 120
cloudhsmv2, 123
cloudsearch, 125
cloudsearchdomain, 128
cloudtrail, 131
cloudtraildataservice, 134
cloudwatch, 136
cloudwatchevents, 140
cloudwatchevidently, 143
cloudwatchinternetmonitor, 146
cloudwatchlogs, 149
cloudwatchobservabilityaccessmanager, 154
cloudwatchrum, 157
codeartifact, 159
codebuild, 166
codecatalyst, 169
codecommit, 174
codedeploy, 181
codeguruprofiler, 186
codegurureviewer, 189
codegurusecurity, 191
codepipeline, 194
codestar, 199
codestarconnections, 202
codestarnotifications, 206
cognitoidentity, 209
cognitoidentityprovider, 212
cognitosync, 217
commit_transaction, 510, 703
compare_faces, 721
complete_attachment_upload, 252
complete_layer_upload, 333, 337
complete_lifecycle_action, 71
complete_migration, 350
complete_multipart_read_set_upload, 604
complete_multipart_upload, 418, 757
complete_snapshot, 313
complete_vault_lock, 418
compose_environments, 354
comprehend, 220
comprehendmedical, 224
computeoptimizer, 227
configservice, 230
configure_agent, 188
configure_health_check, 363
confirm_connection, 288
confirm_customer_agreement, 288
confirm_device, 215
confirm_forgot_password, 215
confirm_private_virtual_interface, 288
confirm_product_instance, 317
confirm_public_virtual_interface, 288
confirm_subscription, 836
create_accelerator, 422
costandusagereportservice, 259
costexplorer, 262
count_closed_workflow_executions, 880
count_open_workflow_executions, 880
count_pending_activity_tasks, 880
count_pending_decision_tasks, 880
create_account, 627
create_account_alias, 445
create_account_assignment, 859
create_account_customization, 688
create_account_link_invitation, 942
create_account_subscription, 688
create_access_control_configuration, 488
create_access_entry, 346
create_access_grant, 761
create_access_grants_instance, 761
create_access_grants_location, 761
create_access_key, 445
create_access_log_subscription, 909
create_access_point, 343, 761
create_access_point_for_object_lambda, 761
create_access_policy, 615
create_access_preview, 10
create_access_token, 173
create_account, 627
create_account_alias, 445
create_account_assignment, 859
create_account_customization, 688
create_account_link_invitation, 942
create_account_subscription, 688
create_acl, 575
create_action, 769
create_action_target, 799
create_activation, 843
create_activity, 828
create_adapter, 889
create_adapter_version, 889
create_additional_assignments_for_hit, 580
create_account, 346
create_agent_status, 237
create_account, 551
create_alert_manager_definition, 673
create_algorithm, 769
create_alias, 291, 507, 514, 633, 935
create_allow_list, 557
create_analysis, 688
create_analyzer, 10
create_annotation_store, 604
create_annotation_store_version, 604
create_anomaly_detector, 551
create_anomaly_monitor, 264
create_anomaly_subscription, 264
create_api, 28
create_api_destination, 142, 382
create_api_key, 21, 922
create_api_mapping, 28
create_app, 618, 652, 725, 769
create_app_authorization, 32
create_app_block, 53
create_app_block_builder, 53
create_app_block_builder_streaming_url, 53
create_app_bundle, 32
create_app_cookie_stickiness_policy, 363
create_app_image_config, 769
create_app_monitor, 159
create_app_version_app_component, 725
create_app_version_resource, 725
create_application, 40, 46, 53, 184, 354, 376, 500, 503, 807, 859
create_application_assignment, 859
create_application_instance, 630
create_application_presigned_url, 503
create_application_snapshot, 503
create_application_version, 354, 807
create_approval_rule_template, 179
create_archive, 142, 382
create_archive_rule, 10
create_artifact, 769
create_assessment, 65
create_assessment_framework, 65
create_assessment_report, 65
create_assessment_target, 460
create_assessment_template, 460
create_asset, 273
create_asset_revision, 273
create_asset_type, 273
create_assistant, 254
create_assistant_association, 254
create_association, 843
create_association_batch, 843
create_attribute_group, 46
create_authentication_profile, 708
create_authorizer, 21, 28
create_auto_ml_job, 769
create_auto_ml_job_v2, 769
create_auto_predictor, 407
create_auto_scaling_configuration, 49
create_auto_scaling_group, 71
create_automation_rule, 800
create_availability_configuration, 935
create_aws_log_source, 804
create_backup, 307, 415, 623
create_backup_plan, 77
create_backup_selection, 77
create_backup_vault, 77
create_base_path_mapping, 21
create_batch_import_job, 411
create_batch_inference_job, 641
create_batch_load_task, 894
create_batch_prediction, 554
create_batch_prediction_job, 411
create_batch_segment_job, 641
create_bgp_peer, 288
create_billing_group, 95
create_blue_green_deployment, 697
create_blueprint, 426
create_bot, 521
create_bot_alias, 521
create_bot_locale, 521
create_bot_replica, 521
create_bot_version, 518, 521
create_branch, 179
create_broker, 577
create_browser_settings, 946
create_bucket, 539, 757, 761
create_bucket_access_key, 539
create_budget, 101
create_budget_action, 101
create_byte_match_set, 913, 917
create_cache_cluster, 350
create_cache_security_group, 350
create_cache_subnet_group, 350
create_cakedi_scsi_volume, 867
create_calculated_attribute_definition, 267
create_call_analytics_category, 897
create_campaign, 244, 641, 652
create_canary, 883
create_capacity_provider, 339
create_capacity_reservation, 60, 317
create_capacity_reservation_fleet, 317
create_carrier_gateway, 317
create_case, 246, 874
create_cell, 750
create_certificate, 539
create_certificate_authority, 18
create_certificate_authority_audit_report, 18
create_change_set, 113
create_changeset, 394
create_channel, 133, 470
create_chat_token, 475
create_cidr_collection, 737
create_cis_scan_configuration, 464
create_classification_job, 557
create_classifier, 426
create_cli_token, 584
create_client_vpn_endpoint, 317
create_client_vpn_route, 317
create_cloud_formation_change_set, 807
create_cloud_formation_stack, 539
create_cloud_formation_template, 807
create_cloud_front_origin_access_identity, 117
create_cluster, 125, 278, 301, 339, 346, 483, 575, 708, 747, 769
create_cluster_parameter_group, 708
create_cluster_security_group, 708
create_cluster_snapshot, 301, 708
create_cluster_subnet_group, 708
create_cluster_v2, 483
create_code_repository, 769
create_code_review, 191
create_code_signing_config, 514
create_coip_cidr, 317
create_coip_pool, 317
create_collection, 615, 721
create_comment, 929
create_commit, 180
create_compilation_job, 769
create_component, 40, 457, 678
create_compute_environment, 87
create_computer, 291
create_conditional_forwarder, 291
create_configuration, 483, 578
create_configuration_policy, 800
create_configuration_set, 658, 661, 664, 820, 823
create_configuration_set_event_destination, 658, 661, 820, 823
create_configuration_set_tracking_options, 820
create_configuration_template, 354
create_connect_attachment, 597
create_connect_client_add_in, 942
create_connect_peer, 597
create_connection, 49, 142, 205, 288, 382, 426, 597
create_connection_alias, 942
create_connector, 486, 639
create_constraints, 810
create_contact, 823, 848
create_contact_channel, 848
create_contact_flow, 237
create_contact_flow_module, 237
create_contact_list, 823
create_contact_method, 539
create_container_recipe, 457
create_container_service, 539
create_container_service_deployment, 539
create_container_service_registry_login, 539
create_content, 254
create_context, 769
create_continuous_deployment_policy, 117
create_control, 65
create_control_panel, 747
create_core_network, 597
create_cost_category_definition, 264
create_crawler, 426
create_cross_account_attachment, 422
create_cross_account_authorization, 750
create_custom_action_type, 198
create_custom_data_identifier, 557
create_custom_db_engine_version, 697
create_custom_domain_association, 708, 716
create_custom_entity_type, 426
create_custom_key_store, 507
create_custom_line_item, 95
create_custom_log_source, 804
create_custom_metadata, 929
create_custom_plugin, 486
create_custom_routing_accelerator, 422
create_custom_routing_endpoint_group, 422
create_custom_routing_listener, 422
create_custom_verification_email_template, 820, 823
create_customer_gateway, 317
create_dashboard, 688
create_data_catalog, 60
create_data_cells_filter, 510
create_data_lake, 804
create_data_lake_exception_subscription, 804
create_data_lake_organization_configuration, 804
create_data_quality_job_definition, 769
create_data_quality_ruleset, 426
create_data_repository_association, 415
create_data_repository_task, 415
create_data_set, 688
create_data_source, 273, 489, 688
create_data_source_from_rds, 554
create_data_source_from_redshift, 554
create_data_source_from_s3, 554
create_data_view, 394
create_database, 426, 894
create_dataset, 222, 394, 407, 432, 548, 641, 721
create_dataset_export_job, 641
create_dataset_group, 407, 641
create_dataset_import_job, 407, 641
create_db_cluster, 297, 586, 697
create_db_cluster_endpoint, 586, 697
create_db_cluster_parameter_group, 297, 586, 697
create_db_cluster_snapshot, 297, 586, 697
create_db_instance, 297, 586, 697
create_db_instance_read_replica, 697
create_db_parameter_group, 586, 697
create_db_proxy, 697
create_db_proxy_endpoint, 697
create_db_security_group, 697
create_db_shard_group, 697
create_db_snapshot, 697
create_db_subnet_group, 297, 586, 697
create_dedicated_ip_pool, 658, 823
create_default_subnet, 317
create_default_vpc, 317
create_deliverability_test_report, 658, 824
create_delivery, 152
create_delivery_stream, 397
create_deployment, 21, 28, 184, 618
create_deployment_config, 184
create_deployment_group, 184
create_detector, 436
create_detector_version, 411
create_dev_endpoint, 426
create_dev_environment, 173
create_device, 597
create_device_fleet, 769
create_dhcp_options, 317
create_direct_connect_gateway, 288
create_direct_connect_gateway_association, 288
create_direct_connect_gateway_association_proposal, 288
create_directory, 109, 291
create_directory_config, 53
create_directory_registration, 639
create_discoverer, 792
create_disk, 539
create_disk_from_snapshot, 539
create_disk_snapshot, 539
create_distribution, 117, 539
create_distribution_configuration, 457
create_distribution_with_tags, 117
create_document, 843
create_document_classifier, 222
create_documentation_part, 21
create_documentation_version, 21
create_domain, 127, 164, 246, 267, 274, 539, 611, 769, 834, 906
create_domain_entry, 539
create_domain_name, 21, 28
create_edge_deployment_plan, 769
create_edge_deployment_stage, 769
create_edge_packaging_job, 769
create_egress_only_internet_gateway, 317
create_eks_anywhere_subscription, 346
create_elasticsearch_domain, 360
create_email_identity, 658, 824
create_email_identity_policy, 824
create_email_template, 652, 824
create_encoder_configuration, 480
create_endpoint, 222, 382, 765, 769
create_endpoint_access, 708, 716
create_endpoint_config, 769
create_endpoint_group, 422
create_entitlement, 53
create_entity_recognizer, 222
create_environment, 274, 354, 391, 584, 678
create_environment_account_connection, 678
create_environment_ec2, 102, 104
create_environment_membership, 102, 104
create_environment_profile, 274, 354, 391, 584, 678
create_environment_template, 678
create_environment_template_version, 678
create_evaluation, 554
create_evaluation_form, 237
create_evaluation_job, 90
create_event_bus, 142, 382
create_event_data_store, 133
create_event_destination, 664
create_event_source_mapping, 514
create_event_stream, 267
create_event_subscription, 297, 587, 697, 708
create_event_tracker, 641
create_exclusions_preview, 460
create_experience, 489
create_experiment, 145, 769
create_experiment_template, 399
create_explainability, 407
create_explainability_export, 407
create_export, 521
create_export_job, 652, 824
create_export_task, 152
create_extended_source_server, 303
create_face_liveness_session, 721
create_facet, 109
create_faq, 489
create_fargate_profile, 346
create_feature, 145
create_feature_group, 769
create_featured_results_set, 489
create_fhir_datastore, 442
create_field, 246
create_field_level_encryption_config, 117
create_field_level_encryption_profile, 117
create_file_cache, 415
create_file_system, 343, 415
create_file_system_from_backup, 415
create_filter, 436, 464, 642
create_findings_aggregator, 800
create_findings_filter, 557
create_findings_report, 464
create_firewall, 594
create_firewall_domain_list, 753
create_firewall_policy, 594
create_firewall_rule, 753
create_firewall_rule_group, 753
create_fleet, 53, 168, 317, 932
create_flow_definition, 769
create_flow_logs, 317
create_flywheel, 222
create_folder, 688, 929
create_folder_membership, 688
create_forecast, 407
create_forecast_export_job, 407
create_form_type, 274
create_fpga_image, 317
create_framework, 77
create_function, 117, 514
create_function_url_config, 514
create_gateway, 81
create_gateway_route, 43
create_generated_template, 113
create_geo_match_set, 913, 917
create_geofence_collection, 545
create_global_cluster, 297, 587, 697, 708
create_global_network, 597
create_global_replication_group, 350
create_global_table, 307
create_global_table, 307
create_glossary, 274
create_glossary_term, 274
create_gov_cloud_account, 627
create_grant, 507, 531
create_grant_version, 531
create_graph, 282
create_group, 215, 445, 454, 688, 732, 883, 935, 949
create_group_membership, 454, 688
create_group_profile, 274
create_guardrail, 90
create_guardrail_version, 90
create_gui_session_access_details, 539
create_hapg, 122
create_health_check, 737
create_hit, 580
create_hit_type, 580
create_hit_with_hit_type, 580
create_host, 205
create_hosted_zone, 737
create_hours_of_operation, 237
create_hsm, 122, 125
create_hsm_client_certificate, 708
create_hsm_configuration, 708
create_http_namespace, 814
create_hub, 769
create_human_task_ui, 769
create_hyper_parameter_tuning_job, 769
create_iam_policy_assignment, 688
create_id_mapping_workflow, 379
create_id_namespace, 379
create_identity_pool, 211
create_identity_provider, 215, 946
create_identity_source, 904
create_image, 317, 457, 769
create_image_builder, 53
create_image_builder_streaming_url, 53
create_image_pipeline, 457
create_image_recipe, 457
create_image_version, 769
create_impersonation_role, 935
create_import_job, 652, 824
create_in_app_template, 652
create_index, 109, 489, 729
create_instance_event_window, 317
create_instance_export_task, 317
create_instance_profile, 445
create_instance_snapshot, 445
create_instances, 539
create_instances_from_snapshot, 539
create_integration, 28, 697
create_integration_association, 237
create_integration_response, 28
create_integration_workflow, 267
create_intent, 521
create_intent_version, 518
create_interconnect, 288
create_internet_gateway, 317
create_invalidation, 117
create_invitations, 557
create_ip_access_settings, 946
create_ip_group, 942
create_ip_set, 436, 913, 917, 922
create_ipam, 317
create_ipam_pool, 317
create_ipam_resource_discovery, 317
create_ipam_scope, 317
create_job, 98, 426, 761
create_job_for_devices, 630
create_job_queue, 87
create_job_template, 373
create_journey, 652
create_key, 507, 545, 633
create_key_group, 117
create_key_pair, 317, 539
create_key_signing_key, 737
create_key_value_store, 117
create_keyspace, 495
create_knowledge_base, 254
create_kx_changeset, 391
create_kx_cluster, 391
create_kx_database, 391
create_kx_dataview, 391
create_kx_environment, 391
create_kx_scaling_group, 391
create_kx_user, 391
create_kx_volume, 391
create_label, 548
create_label_group, 548
create_labeling_job, 769
create_labels, 929
create_lag, 288
create_lake_formation_identity_center_configuration, 510
create_lake_formation_opt_in, 510
create_landing_zone, 258
create_language_model, 897
create_launch, 145
create_launch_configuration, 71
create_launch_configuration_template, 303
create_launch_profile, 601
create_launch_template, 317
create_launch_template_version, 317
create_layer, 619
create_layout, 246
create_lb_cookie_stickiness_policy, 363
create_ledger, 682
create_legal_hold, 77
create_lens_share, 925
create_lens_version, 925
create_lf_tag, 510
create_license, 531
create_license_configuration, 531
create_license_conversion_task_for_resource, 531
create_license_manager_report_generator, 531
create_license_version, 531
create_lifecycle_policy, 295, 457, 615
create_link, 156, 597
create_list, 411
create_listener, 367, 422, 909
create_listing_change_set, 274
create_load_balancer, 363, 367, 539
create_load_balancer_listeners, 363
create_load_balancer_policy, 363
create_load_balancer_tls_certificate, 539
create_local_gateway_route, 317
create_local_gateway_route_table, 317
create_local_gateway_route_table_virtual_interface_group_association, 317
create_local_gateway_route_table_vpc_association, 317
create_log_anomaly_detector, 152
create_log_group, 152
create_log_pattern, 40
create_log_stream, 152
create_log_subscription, 291
create_logging_configuration, 475, 673
create_logically_air_gapped_backup_vault, 77
create_login_profile, 445
create_luna_client, 122
create_maintenance_window, 843
create_managed_endpoint, 373
create_managed_prefix_list, 317
create_map, 545
create_matching_workflow, 379
create_medical_vocabulary, 897
create_member, 557
create_members, 282, 436, 800
create_mesh, 43
create_microsoft_ad, 291
create_milestone, 925
create_ml_endpoint, 590
create_ml_model, 554
create_ml_transform, 426
create_mobile_device_access_rule, 936
create_model, 21, 28, 411, 548, 769
create_model_bias_job_definition, 769
create_model_card, 769
create_model_card_export_job, 769
create_model_customization_job, 90
create_model_explainability_job_definition, 769
create_model_package, 769
create_model_package_group, 769
create_model_quality_job_definition, 769
create_model_template, 411
create_monitor, 149, 407
create_monitoring_schedule, 769
create_monitoring_subscription, 117
create_multipurpose_vault, 761
create_multipurpose_vault_iam_policy, 761
create_multipurpose_vault_policy, 761
create_multiregion_access_point_iam_policy, 761
create_multiregion_access_point_policy, 761
create_multiregion_access_point_vpc_association, 761
create_multipart_read_set_upload, 604
create_multipart_upload, 757
create_named_query, 60
create_namespace, 688, 716
create_nat_gateway, 317
create_network_acl, 317
create_network_acl_entry, 317
create_network_insights_access_scope,
create_network_insights_path, 317
create_network_interface, 317
create_network_interface_permission, 317
create_network_settings, 946
create_nfs_file_share, 867
create_node_from_template_job, 630
create_nodegroup, 346
create_notebook, 60
create_notebook_instance, 769
create_notebook_instance_lifecycle_config, 769
create_notification, 101
create_notification_rule, 208
create_object, 109
create_observability_configuration, 49
create_open_id_connect_provider, 445
create_ops_item, 843
create_ops_metadata, 843
create_opt_out_list, 664
create_option_group, 698
create_parallel_data, 900
create_parameter_group, 278, 575
create_participant, 237
create_participant_connection, 252
create_participant_token, 480
create_partition, 426
create_partition_index, 426
create_partner_event_source, 142, 382
create_patch_baseline, 843
create_performance_analysis_report, 650
create_permission, 18, 693
create_permission_group, 394
create_permission_set, 859
create_permission_version, 859
create_persistent_contact_association, 237
create_pipe, 386
create_pipeline, 198, 271, 608, 769
create_place_index, 545
create_placement_group, 317
create_platform_application, 836
create_platform_endpoint, 836
create_platform_version, 354
create_playback_restriction_policy, 470
create_pod_identity_association, 346
create_policy, 445, 627, 904
create_policy_store, 904
create_policy_template, 904
create_policy_version, 445
create_pool, 664
create_portal, 946
create_portfolio, 810
create_portfolio_share, 810
create_practice_run_configuration, 57
create_predefined_attribute, 237
create_predictor, 407
create_predictor_backtest_export_job, 407
create_prepared_statement, 60
create_presigned_domain_url, 769
create_presigned_notebook_instance_url, 770
create_presigned_notebook_url, 60
create_pricing_plan, 95
create_pricing_rule, 95
create_private_dns_namespace, 814
create_private_virtual_interface, 288
create_processing_job, 770
create_product, 810
create_profile, 267, 451, 925
create_profile_job, 432
create_profile_share, 925
create_profiling_group, 188
create_project, 145, 168, 173, 201, 274, 432, 721, 770
create_project_membership, 274
create_project_version, 721
create_prompt, 237
create_protect_configuration, 664
create_protection, 831
create_protection_group, 831
create_provisioned_model_throughput, 90
create_provisioned_product_plan, 810
create_provisioning_artifact, 810
create_public_dns_namespace, 814
create_public_ipv_4_pool, 317
create_public_key, 117
create_public_virtual_interface, 288
create_publishing_destination, 436
create_pull_request, 180
create_pull_request_approval_rule, 180
create_pull_through_cache_rule, 333
create_push_template, 652
create_qualification_type, 580
create_quantum_task, 98
create_query_logging_config, 737
create_query_suggestions_block_list, 489
create_queue, 237, 840
create_quick_connect, 237
create_quick_response, 254
create_rate_based_rule, 913, 917
create_readiness_check, 750
create_realtime_endpoint, 554
create_realtime_log_config, 117
create_receipt_filter, 820
create_receipt_rule, 820
create_receipt_rule_set, 820
create_recipe, 432
create_recipe_job, 432
create_recommendation_template, 725
create_recommender, 642
create_recommender_configuration, 653
create_recording_configuration, 470
create_recovery_group, 750
create_redshift_idc_application, 708
create_reference_store, 604
create_refresh_schedule, 688
create_regex_match_set, 913, 917
create_regex_pattern_set, 913, 917, 922
create_registration, 664
create_registration_association, 664
create_registration_attachment, 664
create_registration_version, 664
create_registry, 426, 792
create_related_item, 246
create_relational_database, 539
create_relational_database_from_snapshot, 539
create_relational_database_snapshot, 539
create_replace_root_volume_task, 317
create_replication_configuration, 343
create_replication_configuration_template, 303
create_replication_group, 350
create_replication_set, 851
create_replicator, 483
create_report_group, 168
create_report_plan, 77
create_repository, 164, 180, 333, 337, 678
create_repository_link, 205
create_request_validator, 21
create_rescore_execution_plan, 492
create_reserved_instances_listing, 318
create_resiliency_policy, 725
create_resolover_endpoint, 753
create_resolover_query_log_config, 753
create_resolover_rule, 753
create_resource, 21, 107, 936
create_resource_data_sync, 843
create_resource_group, 460
create_resource_policy, 521
create_resource_policy_statement, 521
create_resource_server, 215
create_resource_set, 750
create_resource_share, 694
create_response_headers_policy, 117
create_response_plan, 851
create_rest_api, 21
create_restore_image_task, 318
create_restore_testing_plan, 77
create_restore_testing_selection, 77
create_retraining_scheduler, 548
create_reusable_delegation_set, 737
create_review_template, 925
create_role, 445
create_role_membership, 688
create_room, 475
create_rotation, 848
create_rotation_override, 848
create_route, 28, 43, 318
create_route_calculator, 545
create_route_response, 28
create_route_table, 318
create_routing_control, 747
create_routing_profile, 237
create_rule, 237, 367, 411, 705, 909, 913, 917
create_rule_group, 594, 913, 917, 922
create_rule_groups_namespace, 673
create_ruleset, 432
create_run_group, 604
create_safety_rule, 747
create_saml_provider, 445
create_sampling_rule, 949
create_savings_plan, 790
create_sbom_export, 464
create_scaling_plan, 75
create_scan, 193
create_schedule, 388, 432
create_schedule_group, 388
create_scheduled_action, 708, 716
create_scheduled_query, 892
create_scheduling_policy, 87
create_schema, 109, 426, 642, 792
create_schema_mapping, 379
create_scraper, 673
create_script, 426
create_secret, 795
create_security_config, 615
create_security_configuration, 370, 373, 426
create_security_group, 318
create_security_policy, 615
create_security_profile, 237
create_segment, 145, 653
create_sequence_store, 604
create_server, 623
create_serverless_cache, 350
create_serverless_cache_snapshot, 350
create_service, 50, 339, 678, 814, 909
create_service_action, 810
create_service_instance, 678
create_service_linked_role, 445
create_service_network, 909
create_service_network_service_association, 909
create_service_network_vpc_association, 909
create_service_principal_name, 639
create_service_specific_credential, 445
create_service_sync_config, 678
create_service_template, 678
create_service_template_version, 678
create_session, 254, 426, 757
create_share, 604
create_sink, 156
create_site, 597
create_site_to_site_vpn_attachment, 597
create_size_constraint_set, 913, 917
create_slack_channel_configuration, 877
create_slot, 521
create_slot_type, 522
create_slot_type_version, 518
create_smb_file_share, 867
create_sms_sandbox_phone_number, 836
create_sms_template, 653
create_snapshot, 291, 318, 350, 415, 575, 716, 867
create_snapshot_copy_configuration, 716
create_snapshot_copy_grant, 708
create_snapshot_from_volume_recovery_point, 867
create_snapshot_schedule, 708
create_snapshots, 318
create_sol_function_package, 886
create_sol_network_instance, 886
create_sol_network_package, 886
create_solution, 642
create_solution_version, 642
create_source_network, 303
create_source_repository, 173
create_source_repository_branch, 173
create_space, 770
create_spot_datafeed_subscription, 318
create_sql_injection_match_set, 913, 917
create_stack, 53, 113, 619
create_stack_instances, 113
create_stack_set, 113
create_stage, 21, 28, 480
create_standby_workspaces, 942
create_state_machine, 828
create_state_machine_alias, 828
INDEX

create_storage_configuration, 480
create_storage_lens_group, 761
create_storage_location, 354
create_storage_virtual_machine, 415
create_store_image_task, 318
create_storedi_scsi_volume, 867
create_stream, 497
create_stream_key, 470
create_stream_processor, 721
create_streaming_distribution, 117
create_streaming_distribution_with_tags, 118
create_streaming_image, 601
create_streaming_session, 601
create_streaming_session_stream, 601
create_streaming_url, 53
create_studio, 370, 601
create_studio_component, 601
create_studio_lifecycle_config, 770
create_studio_session_mapping, 370
create_subnet, 318
create_subnet_cidr_reservation, 318
create_subnet_group, 278, 575
create_subscriber, 101, 804
create_subscriber_notification, 804
create_subscription, 831
create_subscription_grant, 274
create_subscription_request, 274
create_subscription_target, 274
create_sync_configuration, 205
create_table, 307, 426, 495, 894
create_table_optimizer, 426
create_tag_option, 810
create_tags, 318, 343, 578, 708, 942
create_tape_pool, 867
create_tape_with_barcode, 867
create_tapes, 867
create_target_account_configuration, 399
create_target_group, 367, 909
create_task_set, 339
create_task_template, 237
create_template, 246, 639, 688, 820
create_template_alias, 688
create_template_group_access_control_entry, 639
create_template_share, 925
create_template_sync_config, 678
create_tenant_database, 698
create_test_set_discrepancy_report, 522
create_theme, 688
create_theme_alias, 688
create_thesaurus, 489
create_threat_intel_set, 436
create_timeline_event, 851
create_tls_inspection_configuration, 594
create_token, 531, 863
create_token_with_iam, 864
create_topic, 688, 836
create_topic_refresh_schedule, 688
create_tracker, 545
create_traffic_distribution_group, 237
create_traffic_mirror_filter, 318
create_traffic_mirror_filter_rule, 318
create_traffic_mirror_session, 318
create_traffic_mirror_target, 318
create_traffic_policy, 737
create_traffic_policy_instance, 737
create_traffic_policy_version, 737
create_trail, 133
create_training_job, 770
create_transform_job, 770
create_transit_gateway, 318
create_transit_gateway_connect, 318
create_transit_gateway_connect_peer, 318
create_transit_gateway_multicast_domain, 318
create_transit_gateway_peering, 597
create_transit_gateway_peering_attachment, 318
create_transit_gateway_policy_table, 318
create_transit_gateway_prefix_list_reference, 318
create_transit_gateway_route, 318
create_transit_gateway_route_table, 318
create_transit_gateway_route_table_announcement, 318
create_transit_gateway_route_table_attachment, 597
create_transit_gateway_vpc_attachment, 318
create_transit_virtual_interface, 288
create_trial, 770
create_trial_component, 770
create_trigger, 426
create_trust, 291
create_trust_anchor, 451
create_trust_store, 367, 946
create_trusted_token_issuer, 860
create_typed_link_facet, 109
create_unreferenced_merge_commit, 180
create_updated_image, 53
create_updated_workspace_image, 942
create_upload_url, 193, 522
create_usage_limit, 708, 716
create_usage_plan, 21
create_usage_plan_key, 21
create_usage_report_subscription, 53
create_use_case, 237
create_user, 33, 237, 350, 445, 454, 575, 578, 721, 929, 936
create_user_access_logging_settings, 946
create_user_defined_function, 426
create_user_group, 350
create_user_hierarchy_group, 237
create_user_import_job, 215
create_user_pool, 215
create_user_pool_client, 215
create_user_pool_domain, 215
create_user_profile, 201, 274, 619, 770
create_user_settings, 946
create_variable, 411
create_variant_store, 605
create_vault, 418
create_verified_access_endpoint, 318
create_verified_access_group, 318
create_verified_access_instance, 318
create_verified_access_trust_provider, 318
create_verified_destination_number, 664
create_view, 237, 729
create_view_version, 237
create_virtual_cluster, 373
create_virtual_gateway, 43
create_virtual_mfa_device, 445
create_virtual_node, 43
create_virtual_router, 43
create_virtual_service, 43
create_vocabulary, 237, 897
create_vocabulary_filter, 897
create_voice_template, 653
create_volume, 318, 415
create_volume_from_backup, 415
create_vpc, 318
create_vpc_association_authorization, 737
create_vpc_attachment, 597
create_vpc_connection, 483, 688
create_vpc_connector, 50
create_vpc_endpoint, 318, 360, 611, 615
create_vpc_endpoint_connection_notification, 318
create_vpc_endpoint_service_configuration, 318
create_vpc_ingress_connection, 50
create_vpc_link, 21, 28
create_vpc_peering_connection, 318
create_vpn_connection, 318
create_vpn_connection_route, 318
create_vpn_gateway, 318
create_web_acl, 906
create_web_acl_migration_stack, 913, 917
create_web_login_token, 584
create_webhook, 168
create_what_if_analysis, 407
create_what_if_forecast, 407
create_what_if_forecast_export, 407
create_work_group, 60
create_worker_block, 580
create_worker_configuration, 486
create_workflow, 426, 457, 605
create_workforce, 770
create_workgroup, 716
create_workload, 925
create_workload_share, 925
create_workspace, 561, 673
create_workspace_api_key, 561
create_workspace_bundle, 942
create_workspace_image, 942
create_workspaces, 942
create_workteam, 770
create_xss_match_set, 913, 917
customerprofiles, 265
RESTART INDEX

datapipeline, 268
datazone, 271
dax, 276
deactivate_anomaly_detector, 551
deactivate_contact_channel, 848
deactivate_evaluation_form, 237
deactivate_event_source, 142, 382
deactivate_key_signing_key, 737
deactivate_mfa_device, 445
deactivate_organizations_access, 113
deactivate_type, 113
deactivate_user, 929
deauthorize_connection, 142, 382
deauthorize_data_share, 708
decline_handshake, 627
decline_invitations, 436, 557, 800
decode_authorization_message, 871
decrease_node_groups_in_global_replication_group, 350
decrease_replica_count, 350
decrease_replication_factor, 278
decrease_stream_retention_period, 497
decrypt, 507
decrypt_data, 636
define_analysis_scheme, 127
define_expression, 127
define_index_field, 127
define_suggester, 127
delete_accelerator, 422
delete_access_control_configuration, 489
delete_access_control_rule, 936
delete_access_entry, 346
delete_access_grant, 761
delete_access_grants_instance, 761
delete_access_grants_instance_resource_policy, 761
delete_access_grants_location, 761
delete_access_key, 445
delete_access_log_settings, 28
delete_access_log_subscription, 909
delete_access_point, 343, 761
delete_access_point_for_object_lambda, 761
delete_access_point_policy, 761
delete_access_point_policy_for_object_lambda, 761
delete_access_policy, 615
delete_access_token, 173
delete_account_alias, 445, 878
delete_account_assignment, 860
delete_account_customization, 688
delete_account_default_protect_configuration, 664
delete_account_link_invitation, 942
delete_account_password_policy, 445
delete_account_policy, 152
delete_account_setting, 339
delete_account_subscription, 688
delete_acl, 575
delete_action, 770
delete_action_target, 800
delete_activation, 843
delete_activity, 828
delete_adapter, 889
delete_adapter_version, 889
delete_addon, 346
delete_adm_channel, 653
delete_aggregation_authorization, 232
delete_alarm, 539
delete_alerts, 139
delete_alert, 551
delete_alert_manager_definition, 673
delete_algorithm, 770
delete_alias, 507, 514, 633, 936
delete_allow_list, 557
delete_alternate_contact, 13
delete_analysis, 688
delete_analysis_scheme, 127
delete_analyzer, 10
delete_annotation_store, 605
delete_annotation_store_versions, 605
delete_anomaly_detector, 139, 551
delete_anomaly_monitor, 264
delete_anomaly_subscription, 264
delete_api, 28
delete_api_destination, 142, 382
delete_api_key, 21, 922
delete_api_mapping, 28
delete_apns_channel, 653
delete_apns_sandbox_channel, 653
delete_apns_voip_channel, 653
delete_apns_voip_sandbox_channel, 653
delete_app, 619, 653, 725, 770
delete_app_assessment, 725
delete_app_authorization, 32
delete_app_block, 53
delete_app_block_builder, 53
delete_app_bundle, 32
delete_app_image_config, 770
delete_app_input_source, 725
delete_app_monitor, 159
delete_app_version_app_component, 725
delete_app_version_resource, 725
delete_application, 40, 46, 53, 500, 503, 807, 860
delete_application_access_scope, 860
delete_application_assignment, 860
delete_application_authentication_method, 860
delete_application_cloud_watch_logging_option, 500, 503
delete_application_grant, 860
delete_application_input_processing_configuration, 500, 503
delete_application_output, 500, 503
delete_application_reference_data_source, 500, 503
delete_application_snapshot, 503
delete_application_version, 354
delete_application_vpc_configuration, 503
delete_approval_rule_template, 180
delete_apps_list, 402
delete_archive, 142, 382, 418
delete_archive_rule, 10
delete_artifact, 770
delete_assessment, 65
delete_assessment_framework, 65
delete_assessment_framework_share, 65
delete_assessment_report, 65
delete_assessment_run, 460
delete_assessment_target, 460
delete_assessment_template, 460
delete_asset, 274
delete_asset_type, 274
delete_assistant, 254
delete_assistant_association, 254
delete_association, 770, 843
delete_attachment, 597
delete_attribute_group, 46
delete_attribute_mapping, 451
delete_attributes, 339, 834

delete_auth_policy, 909
delete_authentication_profile, 709
delete_authorizer, 21, 28
delete_auto_scaling_configuration, 50
delete_auto_scaling_group, 71
delete_auto_snapshot, 540
delete_automatic_tape_creation_policy, 867
deleteAvailability_configuration, 936
delete_aws_log_source, 804
delete_backup, 125, 307, 415, 623
delete_backup_plan, 77
delete_backup_selection, 77
delete_backup_vault, 77
delete_backup_vault_access_policy, 77
delete_backup_vault_notifications, 77
delete_baidu_channel, 653
delete_bandwidth_rate_limit, 867
delete_base_path_mapping, 21
delete_batch_import_job, 411
delete_batch_prediction, 554
delete_batch_prediction_job, 411
delete_bgp_peer, 288
delete_billing_group, 95
delete_blue_green_deployment, 698
delete_blueprint, 426
delete_bot, 518, 522
delete_bot_alias, 518, 522
delete_bot_channel_association, 518
delete_bot_locale, 522
delete_bot_replica, 522
delete_bot_version, 518, 522
delete_branch, 180
delete_broker, 578
delete_browser_settings, 946
delete_bucket, 540, 757, 761
delete_bucket_access_key, 540
delete_bucket_analytics_configuration, 757
delete_bucket_cors, 757
delete_bucket_encryption, 757
delete_bucket_intelligent_tiering_configuration, 757
delete_bucket_inventory_configuration, 757
delete_bucket_lifecycle, 757
delete_bucket_lifecycle_configuration, 761
delete_bucket_metrics_configuration, 757
delete_bucket_ownership_controls, 757
delete_bucket_policy, 757, 761
delete_bucket_replication, 757, 761
delete_bucket_tagging, 757, 761
delete_bucket_website, 757
delete_bucket, 101
delete_budget_action, 101
delete_build_batch, 168
delete_byte_match_set, 913, 917
delete_cache_cluster, 350
delete_cache_parameter_group, 350
delete_cache_policy, 118
delete_cache_security_group, 350
delete_cache_subnet_group, 350
delete_calculated_attribute_definition, 267
delete_call_analytics_category, 897
delete_call_analytics_job, 897
delete_campaign, 244, 642, 653
delete_canary, 883
delete_capacity_provider, 339
delete_capacity_reservation, 60
delete_carrier_gateway, 318
delete_cell, 750
delete_certificate, 15, 540
delete_certificate_authority, 18
delete_change_set, 113
delete_channel, 133, 470
delete_chap_credentials, 867
delete_cidr_collection, 737
delete_cis_scan_configuration, 464
delete_classifier, 426
delete_client_branding, 942
delete_client_certificate, 21
delete_client_vpn_endpoint, 318
delete_client_vpn_route, 318
delete_cloud_front_origin_access_identity, 118
delete_cluster, 125, 278, 301, 339, 346, 483, 575, 709, 747, 770
delete_cluster_subnet_group, 709
delete_cluster_policy, 483
delete_cluster_security_group, 709
delete_cluster_snapshot, 301, 709
delete_cluster_subnet_group, 709
delete_code_repository, 770
delete_code_signing_config, 515
delete_coip_cidr, 318
delete_coip_pool, 318
delete_collection, 615, 721
delete_column_statistics_for_partition, 426
delete_column_statistics_for_table, 426
delete_comment, 929
delete_comment_content, 180
delete_compilation_job, 770
delete_component, 40, 457, 678
delete_compute_environment, 87
delete_conditional_forwarder, 292
delete_config_rule, 232
delete_configuration, 483
delete_configuration_aggregator, 232
delete_configuration_policy, 800
delete_configuration_recorder, 232
delete_configuration_set, 658, 661, 664, 820, 824
delete_configuration_set_event_destination, 658, 661, 820, 824
delete_configuration_set_tracking_options, 820
delete_conformance_template, 354
delete_conformance_pack, 232
delete_connect_client_add_in, 942
delete_connect_instance_config, 244
delete_connect_peer, 597
delete_connection, 25, 50, 142, 205, 288, 382, 427, 597
delete_connection_alias, 942
delete_connector, 486, 639
delete_constraint, 810
delete_contact, 824, 848
delete_contact_channel, 848
delete_contact_evaluation, 237
delete_contact_flow, 237
delete_contact_flow_module, 237
delete_contact_list, 824
delete_contact_method, 540
delete_container_image, 540
delete_container_recipe, 457
delete_container_service, 540
delete_content, 254
delete_context, 770
delete_continuous_deployment_policy, 118
delete_control, 65
delete_control_panel, 747
delete_core_network, 598
delete_core_network_policy_version, 598
delete_cors_configuration, 28
delete_cost_category_definition, 264
delete_crl, 451
delete_cross_account_attachment, 422
delete_cross_account_authorization, 750
delete_custom_action_type, 198
delete_custom_data_identifier, 557
delete_custom_db_engine_version, 698
delete_custom_domain_association, 709, 716
delete_custom_entity_type, 427
delete_custom_key_store, 507
delete_custom_line_item, 95
delete_custom_log_source, 804
delete_custom_metadata, 929
delete_custom_model, 90
delete_custom_plugin, 486
delete_custom_routing_accelerator, 422
delete_custom_routing_endpoint_group, 423
delete_custom_routing_listener, 423
delete_custom_verification_email_template, 820, 824
delete_custom_vocabulary, 522
delete_customer_gateway, 318
delete_dashboard, 688
delete_dashboards, 139
delete_data_catalog, 60
delete_data_cells_filter, 510
delete_data_lake, 804
delete_data_lake_exception_subscription, 804
delete_data_lake_organization_configuration, 804
delete_data_protection_policy, 152
delete_data_quality_job_definition, 770
delete_data_quality_ruleset, 427
delete_data_repository_association, 415
delete_data_set, 688
delete_data_set_refresh_properties, 688
delete_data_source, 274, 489, 554, 611, 688
delete_database, 427, 894
delete_dataset, 219, 394, 408, 432, 548, 642, 721
delete_dataset_group, 408, 642
delete_dataset_import_job, 408
delete_db_cluster, 297, 587, 698
delete_db_cluster_automated_backup, 698
delete_db_cluster_endpoint, 587, 698
delete_db_cluster_parameter_group, 297, 587, 698
delete_db_cluster_snapshot, 298, 587, 698
delete_db_instance, 298, 587, 698
delete_db_instance_automated_backup, 698
delete_db_parameter_group, 587, 698
delete_db_proxy, 698
delete_db_proxy_endpoint, 698
delete_db_security_group, 698
delete_db_shard_group, 698
delete_db_snapshot, 698
delete_db_subnet_group, 298, 587, 698
delete_dedicated_ip_pool, 658, 824
delete_default_message_type, 664
delete_default_sender_id, 664
deflete_delivery, 152
deflete_delivery_channel, 232
deflete_delivery_destination, 152
deflete_delivery_destination_policy, 152
deflete_delivery_source, 152
deflete_delivery_stream, 397
deflete_deployment, 21, 28, 678
deflete_deployment_config, 185
deflete_deployment_group, 185
deflete_destination, 152
deflete_detector, 411, 436
deflete_detector_version, 411
deflete_dev_endpoint, 427
deflete_dev_environment, 173
delete_device, 598, 630
delete_device_fleet, 770
delete_dhcp_options, 318
delete_direct_connect_gateway, 288
delete_direct_connect_gateway_association, 288
delete_direct_connect_gateway_association_proposal, 288
delete_directory, 109, 292
delete_directory_config, 53
delete_directory_registration, 639
delete_discoverer, 792
delete_disk, 540
delete_disk_snapshot, 540
delete_distribution, 118, 540
delete_distribution_configuration, 457
delete_document, 843, 929
delete_document_classifier, 222
delete_document_version, 929
delete_documentation_part, 21
delete_documentation_version, 21
delete_domain, 127, 164, 246, 267, 274, 540, 611, 741, 770, 834, 906
delete_domain_entry, 540
delete_domain_name, 21, 28
delete_domain_permissions_policy, 164
delete_earth_observation_job, 782
delete_edge_deployment_plan, 770
delete_edge_deployment_stage, 770
delete_egress_only_internet_gateway, 318
delete_eks_anywhere_subscription, 346
delete_elasticsearch_domain, 360
delete_elasticsearch_service_role, 360
delete_email_channel, 653
delete_email_identity, 638, 824
delete_email_identity_policy, 824
delete_email_monitoring_configuration, 936
delete_email_template, 653, 824
delete_encoder_configuration, 480
delete_endpoint, 222, 382, 653, 765, 770, 836
delete_endpoint_access, 709, 716
delete_endpoint config, 770
delete_endpoint_group, 423
delete_entitlement, 53
delete_entity_recognizer, 222
delete_entity_type, 411
delete_environment, 102, 104, 274, 391, 584, 678
delete_environment_account_connection, 678
delete_environment_blueprint_configuration, 274
delete_environment_configuration, 354
delete_environment_membership, 102, 104
delete_environment_profile, 274
delete_environment_template, 678
delete_environment_template_version, 678
delete_evaluation, 554
delete_evaluation_form, 237
delete_evaluation_results, 232
delete_event, 411
delete_event_bus, 142, 383
delete_event_data_store, 133
delete_event_destination, 664
delete_event_source_mapping, 515
delete_event_stream, 267, 653
delete_event_subscription, 298, 587, 698, 709
delete_event_tracker, 642
delete_event_type, 411
delete_events_by_event_type, 411
delete_experience, 489
delete_experiment, 145, 770
delete_experiment_template, 399
delete_explainability, 408
delete_explainability_export, 408
delete_export, 522
delete_expression, 127
delete_external_model, 412
delete_faces, 721
delete_facet, 109
delete_faq, 489
delete_fargate_profile, 346
delete_feature, 145
delete_feature_group, 770
delete_fhir_datastore, 443
delete_field, 246
delete_field_level_encryption_config, 118
delete_field_level_encryption_profile, 118
delete_file, 180
delete_file_cache, 415
delete_file_share, 867
delete_file_system, 343, 415
delete_file_system_policy, 343
delete_filter, 436, 464, 642
delete_finding_aggregator, 800
delete_findings_filter, 557
delete_firewall, 594
delete_firewall_domain_list, 753
delete_firewall_manager_ruleGroups, 922
delete_firewall_policy, 594
delete_firewall_rule, 753
delete_fleet, 53, 168, 932
delete_fleets, 319
delete_flow_definition, 770
delete_flow_logs, 319
delete_flywheel, 222
delete_folder, 688, 929
delete_folder_contents, 929
delete_folder_membership, 688
delete_forecast, 408
delete_forecast_export_job, 408
delete_form_type, 274
delete_fpga_image, 319
delete_framework, 77
delete_fraudster, 906
delete_function, 118, 515
delete_function_code_signing_config, 515
delete_function_concurrency, 515
delete_function_event_invoke_config, 515
delete_function_url_config, 515
delete_gateway, 81, 867
delete_gateway_response, 21
delete_gateway_route, 43
delete_gcm_channel, 653
delete_generated_template, 113
delete_geo_match_set, 913, 917
delete_geofence_collection, 545
delete_git_hub_account_token, 185
delete_global_cluster, 298, 587, 698
delete_global_network, 598
delete_global_replication_group, 350
delete_glossary, 274
delete_glossary_term, 274
delete_grant, 531
delete_graph, 282
delete_group, 215, 446, 454, 688, 732, 883, 936, 949
delete_group_membership, 454, 688
delete_group_policy, 446
delete_guardrail, 90
delete_hapg, 122
delete_health_check, 737
delete_hit, 580
delete_host, 205
delete_hosted_zone, 737
delete_hours_of_operation, 237
delete_hsm, 122, 125
delete_hsm_client_certificate, 709
delete_hsm_configuration, 709
delete_hub, 770
delete_hub_content, 770
delete_human_loop, 68
delete_human_task_ui, 770
delete_hyper_parameter_tuning_job, 770
delete_hypervisor, 82
delete_iam_policy_assignment, 688
delete_id_mapping_workflow, 379
delete_id_namespace, 379
delete_identities, 211
delete_identity, 820
delete_identity_policy, 820
delete_identity_pool, 211
delete_identity_propagation_config, 688
delete_identity_provider, 215, 946
delete_identity_source, 904
delete_image, 53, 457, 770
delete_image_builder, 53
delete_image_permissions, 53
delete_image_pipeline, 457
delete_image_recipe, 457
delete_image_version, 770
delete_impersonation_role, 936
delete_import, 522
delete_import_job, 254
delete_imported_key_material, 507
delete_in_app_template, 653
delete_inbound_connection, 611
delete_inbound_cross_cluster_search_connection, 360
delete_incident_record, 851
delete_index, 489, 729
delete_index_field, 127
delete_inference_component, 770
delete_inference_experiment, 770
delete_inference_scheduler, 548
delete_infrastructure_configuration, 457
delete_ingestion, 32
delete_ingestion_destination, 32
delete_inline_policy_from_permission_set, 860
delete_insight, 285, 800
delete_insight_rules, 139
delete_instance, 237, 540, 619, 860
delete_instance_access_control_attribute_configuration, 860
delete_instance_connect_endpoint, 319
delete_instance_event_window, 319
delete_instance_onboarding_job, 244
delete_instance_profile, 446
delete_integration, 21, 28, 267, 698
delete_integration_association, 237
delete_integration_response, 21, 28
delete_intent, 518, 522
delete_intent_version, 518
delete_interconnect, 288
delete_inventory, 843
delete_invitations, 436, 557, 800
delete_ip_access_settings, 946
delete_ip_group, 942
delete_ip_set, 436, 913, 917, 922
delete_ipam, 319
delete_ipam_pool, 319
delete_ipam_resource_discovery, 319
delete_ipamscope, 319
delete_item, 307
delete_job, 303, 427, 432
delete_job_queue, 87
delete_job_tagging, 761
delete_job_template, 373
delete_journey, 653
delete_key, 545, 633
delete_key_group, 118
delete_key_pair, 319, 540
delete_key_signing_key, 737
delete_key_value_store, 118
delete_keyspace, 495
delete_keyword, 664
delete_knowledge_base, 254
delete_known_host_keys, 540
delete_kx_cluster, 391
delete_kx_cluster_node, 391
delete_kx_database, 391
delete_kx_dataview, 391
delete_kx_environment, 391
delete_kx_scaling_group, 391
delete_kx_user, 391
delete_kx_volume, 391
delete_label, 412, 548
delete_label_group, 548
delete_labels, 929
delete_lag, 288
delete_lake_formation_identity_center_configuration, 510
delete_lake_formation_opt_in, 510
delete_landing_zone, 258
delete_language_model, 897
delete_launch, 145
delete_launch_action, 303
delete_launch_configuration, 71
delete_launch_configuration_template, 303
delete_launch_profile, 601
delete_launch_profile_member, 601
delete_launch_template, 319
delete_launch_template_versions, 319
delete_layer, 619
delete_layer_version, 515
delete_layout, 246
delete_ledger, 682
delete_lens, 925
delete_lens_share, 925
delete_lexicon, 668
delete_lf_tag, 510
delete_license, 531
delete_license_configuration, 531
delete_license_manager_report_generator, 531
delete_lifecycle_hook, 71
delete_lifecycle_policy, 295, 333, 457, 615
delete_link, 156, 598
delete_list, 412
delete_listener, 367, 423, 909
delete_listing, 274
delete_load_balancer, 363, 367, 540
delete_load_balancer_listeners, 363
delete_load_balancer_policy, 363
delete_load_balancer_tls_certificate, 540
delete_local_gateway_route, 319
delete_local_gateway_route_table, 319
delete_local_gateway_route_table_virtual_interface_group_association, 319
delete_local_gateway_route_table_vpc_association, 319
delete_log_anomaly_detector, 152
delete_log_group, 152
delete_log_pattern, 40
delete_log_stream, 152
delete_log_subscription, 292
delete_logging_configuration, 475, 673, 913, 917, 922
delete_login_profile, 446
delete_luna_client, 122
delete_mailbox_permissions, 936
delete_maintenance_window, 843
delete_managed_endpoint, 373
delete_managed_prefix_list, 319
delete_map, 545
delete_matching_workflow, 379
delete_medical_scribe_job, 897
delete_medical_transcription_job, 897
delete_medical_vocabulary, 897
delete_member, 557
delete_members, 282, 436, 800
delete_mesh, 44
delete_message, 475, 840
delete_message_batch, 840
delete_method, 21
delete_method_response, 21
delete_metric_attribute, 642
delete_metric_filter, 152
delete_metric_stream, 139
delete_ml_endpoint, 590
delete_ml_model, 554
delete_ml_transform, 427
delete_mobile_device_access_override, 936
delete_mobile_device_access_rule, 936
delete_open_id_connect_provider, 446
delete_ops_item, 843
delete_ops_metadata, 843
delete_opt_out_list, 664
delete_optioned_out_number, 664
delete_option_group, 698
delete_organization, 627, 936
delete_organization_config_rule, 232
delete_organization_conformance_pack, 232
delete_organizational_unit, 627
delete_origin_access_control, 118
delete_origin_request_policy, 118
delete_outbound_connection, 611
delete_outbound_cross_cluster_search_connection, 360
delete_outcome, 412
delete_outpost_resolver, 753
delete_package, 164, 360, 611, 630
delete_package_group, 165
delete_package_versions, 165
delete_parallel_data, 900
delete_parameter, 843
delete_parameter_group, 278, 575
delete_parameters, 843
delete_partition, 427
delete_partition_index, 427
delete_partner, 709
delete_partner_event_source, 142, 383
delete_patch_baseline, 843
delete_peering, 598
delete_pending_aggregation_request, 232
delete_performance_analysis_report, 650
delete_permission, 18, 694
delete_permission_group, 394
delete_permission_policy, 913, 917, 922
delete_permission_set, 860
delete_permission_version, 694
delete_permissions_boundary_from_permission_set, 860
delete_pipe, 386
delete_pipeline, 198, 271, 608, 771
delete_place_index, 545
delete_placement_group, 319
delete_platform_application, 837
delete_platform_version, 354
delete_role_permissions_boundary, 446
delete_role_policy, 446
delete_room, 475
delete_rotation, 848
delete_rotation_override, 848
delete_route, 28, 44, 319
delete_route_calculator, 545
delete_route_request_parameter, 28
delete_route_response, 28
delete_route_settings, 28
delete_route_table, 319
delete_rotation_control, 747
delete_rule, 142, 237, 367, 383, 412, 705, 909, 913, 917
delete_rule_group, 594, 913, 917, 922
delete_rule_groups_namespace, 673
delete_ruleset, 432
delete_rum_metrics_destination, 159
delete_safety_rule, 747
delete_saml_provider, 446
delete_sampling_rule, 949
delete_scaling_plan, 75
delete_scaling_policy, 35
delete_schedule, 388, 432
delete_schedule_group, 388
delete_scheduled_action, 35, 71, 709, 716
delete_scheduled_query, 892
delete_scheduling_policy, 87
delete_schema, 109, 427, 642, 792
delete_schema_mapping, 379
delete_schema_version, 792
delete_schema_versions, 427
delete_scraper, 673
delete_secret, 795
delete_security_config, 615
delete_security_configuration, 370, 427
delete_security_group, 319
delete_security_policy, 615
delete_security_profile, 237
delete_segment, 145, 653
delete_sequence_store, 605
delete_server, 623
delete_server_certificate, 446
delete_serverless_cache, 350
delete_serverless_cache_snapshot, 350
delete_service, 50, 340, 678, 814, 910
delete_service_action, 810
delete_service_linked_role, 446
delete_service_network, 910
delete_service_network_service_association, 910
delete_service_network_vpc_association, 910
delete_service_principal_name, 639
delete_service_quota_increase_request_from_template, 817
delete_service_specific_credential, 446
delete_service_sync_config, 678
delete_service_template, 678
delete_service_template_version, 678
delete_session, 427, 526, 528
delete_share, 605
delete_signing_certificate, 446
delete_sink, 156
delete_site, 598
delete_size_constraint_set, 913, 917
delete_slack_channel_configuration, 878
delete_slack_workspace_configuration, 878
delete_slot, 522
delete_slot_type, 518, 522
delete_slot_type_version, 518
delete_sms_channel, 653
delete_sms_sandbox_phone_number, 837
delete_sms_template, 653
delete_snapshot, 292, 319, 350, 415, 575, 716
delete_snapshot_copy_configuration, 716
delete_snapshot_copy_grant, 709
delete_snapshot_schedule, 709, 867
delete_sol_function_package, 886
delete_sol_network_instance, 886
delete_sol_network_package, 886
delete_solution, 642
delete_source_credentials, 168
delete_source_network, 303
delete_source_repository, 173
delete_source_server, 303
delete_space, 173, 771
delete_sparql_statistics, 590
delete_speaker, 906
delete_spot_datafeed_subscription, 319
delete_sql_injection_match_set, 913, 917
delete_ssh_public_key, 446
delete_stack, 53, 113, 619
delete_stack_instances, 113
delete_stack_set, 113
delete_stage, 21, 28, 480
delete_state_machine, 837
delete_state_machine_alias, 828
delete_state_machine_version, 828
delete_storage_configuration, 480
delete_storage_lens_configuration, 761
delete_storage_lens_configuration_tagging, 761
delete_storage_lens_group, 762
delete_storage_virtual_machine, 415
delete_stored_query, 232
delete_stream, 497
delete_stream_key, 470
delete_stream_processor, 721
delete_streaming_distribution, 118
delete_streaming_image, 601
delete_streaming_session, 601
delete_studio, 370, 601
delete_studio_component, 601
delete_studio_lifecycle_config, 771
delete_studio_member, 601
delete_studio_session_mapping, 370
delete_subnet, 319
delete_subnet_cidr_reservation, 319
delete_subnet_group, 278, 575
delete_subscriber, 101, 804
delete_subscriber_notification, 804
delete_subscription, 831
delete_subscription_filter, 152
delete_subscription_grant, 274
delete_subscription_request, 274
delete_subscription_target, 274
delete_suggester, 127
delete_suppressed_destination, 824
delete_sync_configuration, 205
delete_table, 307, 427, 495, 894
delete_table_optimizer, 427
delete_table_version, 427
delete_tag_option, 810
delete_tags, 71, 319, 343, 554, 578, 709,
delete_transit_gateway_multicast_domain, 319
delete_transit_gateway_peering_attachment, 319
delete_transit_gateway_policy_table, 319
delete_transit_gateway_prefix_list_reference, 320
delete_transit_gateway_route, 320
delete_transit_gateway_route_table, 320
delete_transit_gateway_route_table_announcement, 320
delete_vpc, 418
delete_vpc_assocation_authorization, 738
delete_vpc_connection, 483, 689
delete_vpc_connector, 50
delete_vpc_endpoint, 360, 611, 615
delete_vpc_endpoint_connection_notifications, 320
delete_vpc_endpoint_service_configurations, 320
delete_vpc_endpoints, 320
delete_vpc_ingress_connection, 50
delete_vpc_link, 21, 28
delete_vpc_peering_connection, 320
delete_vpn_connection, 320
delete_vpn_gateway, 320
delete_warm_pool, 71
delete_watchlist, 906
delete_web_acl, 913, 917, 922
describe_account_attributes, 320, 354, 623, 664, 698, 709
describe_account_customization, 689
describe_account_health, 285
describe_account_limits, 71, 113, 363, 367, 664
describe_account_modifications, 942
describe_account_overview, 285
describe_account_policies, 152
describe_account_preferences, 343
describe_account_settings, 689, 892
describe_account_subscription, 689
describe_account, 771
describe_action_targets, 800
describe_activations, 843
describe_active_receipt_rule_set, 820
describe_activities, 929
describe_activity, 828
describe_activity_type, 880
describe_addon, 346
describe_addon_configuration, 346
describe_addon_versions, 346
describe_address_transfers, 320
describe_addresses, 320
describe_addresses_attribute, 320
describe_adjustment_types, 71
describe_affected_accounts_for_organization, 440
describe_affected_entities, 440
describe_agent_status, 238
describe_agent_versions, 619
describe_aggregate_compliance_by_config_rules, 232
describe_aggregate_compliance_by_conformance, 232
describe_aggregate_id_format, 320
describe_aggregation_authorizations, 233
describe_alarm_history, 139
describe_alarms, 139
describe_alarms_for_metric, 139
describe_alert, 551
describe_alert_manager_definition, 673
describe_algorithm, 642, 771
describe_all_managed_products, 922
describe_analysis, 689
describe_analysis_definition, 689
describe_analysis_permissions, 689
describe_analysis_schemes, 127
describe_anomaly, 285
describe_anomaly_detection_executions, 551
describe_anomaly_detector, 551
describe_anomaly_detectors, 139
describe_api_destination, 142, 383
describe_app, 725, 771
describe_app_assessment, 725
describe_app_block_builder_app_block_associations, 53
describe_app_block_builders, 53
describe_app_blocks, 53
describe_app_image_config, 771
describe_app_version, 725
describe_app_version_app_component, 725
describe_app_version_resource, 725
describe_app_version_resources_resolution_status, 725
describe_app_version_template, 726
describe_application, 40, 500, 503, 860
describe_application_assignment, 860
describe_application_assessments, 942
describe_application_fleetAssociations, 53
describe_application_instance, 630
describe_application_instance_details, 630
describe_application_provider, 860
describe_application_snapshot, 503
describe_application_version, 503
describe_application_versions, 354
describe_applications, 53, 354, 942
describe_apps, 619
describe_archive, 142, 383
describe_artifact, 771
describe_assessment_runs, 460
describe_assessment_targets, 460
describe_assessment_templates, 460
describe_asset_bundle_export_job, 689
describe_asset_bundle_import_job, 689
describe_association, 843
describe_association_execution_targets, 843
describe_association_executions, 843
describe_attachment, 874
describe_attack, 831
describe_attack_statistics, 831
describe_audit_stream_configuration, 932
describe_authentication_profiles, 709
describe_auto_ml_job, 771
describe_auto_ml_job_v2, 771
describe_auto_predictor, 408
describe_auto_scaling_configuration, 50
describe_auto_scaling_groups, 71
describe_auto_scaling_instances, 71
describe_auto_scaling_notification_types, 71
describe_automation_executions, 843
describe_automation_step_executions, 844
describe_availability_monitor_test, 867
describe_availability_options, 127
describe_availability_zones, 320
describe_available_patches, 844
describe_aws_network_performance_metric_subscriptions, 320
describe_backup, 307
describe_backup_job, 78
describe_backup_policy, 343
describe_backup_vault, 78
describe_backups, 125, 415, 623
describe_bandwidth_rate_limit, 867
describe_bandwidth_rate_limit_schedule, 867
describe_batch_inference_job, 642
describe_batch_load_task, 894
describe_batch_predictions, 554
describe_batch_segment_job, 642
describe_blue_green_deployments, 698
describe_bot, 522
describe_bot_alias, 522
describe_bot_locale, 522
describe_bot_recommendation, 522
describe_bot_replica, 522
describe_bot_resource_generation, 522
describe_bot_version, 522
describe_broker, 578
describe_broker_instance_options, 578
describe_broker_instance_type_options, 578
describe_buckets, 557
describe_budget, 101
describe_budget_action, 101
describe_budget_action_histories, 101
describe_budget_actions_for_account, 101
describe_budget_actions_for_budget, 101
describe_budget_notifications_for_account, 101
describe_budget_performance_history, 101
describe_budgets, 101
describe_bundle_associations, 942
describe_bundle_tasks, 320
describe_byoip_cidrs, 320
describe_cache, 867
describe_cache_clusters, 350
describe_cache_engine_versions, 350
describe_cache_parameter_groups, 350
describe_cache_parameters, 350
describe_cache_security_groups, 350
describe_cache_subnet_groups, 350
describe_caches, 393, 298, 698
describe_campaign, 244, 642
describe_canaries, 883
describe_canaries_last_run, 883
describe_capacity_block_offerings, 320
describe_capacity_providers, 340
describe_capacity_reservation_fleets, 320
describe_capacity_reservations, 320
describe_carrier_gateways, 320
describe_certificates, 15, 292
describe_certificate_authority, 18
describe_certificate_authority_audit_report, 18
describe_certificates, 298, 698
describe_change_set, 113, 565
describe_change_set_hooks, 113
describe_chap_credentials, 867
describe_classic_link_instances, 320
describe_classification_job, 557
describe_client_authentication_settings, 292
describe_client_branding, 942
describe_custom_key_stores, 507
describe_custom_plugin, 486
describe_custom_routing_accelerator, 423
describe_custom_routing_accelerator_attributes, 423
describe_custom_routing_endpoint_group, 423
describe_custom_routing_listener, 423
describe_custom_vocabulary_metadata, 522
describe_customer_gateways, 320
describe_customer_metadata, 288
describe_dashboard, 689
describe_dashboard_definition, 689
describe_dashboard_permissions, 689
describe_dashboard_snapshot_job, 689
describe_dashboard_snapshot_job_result, 689
describe_data_ingestion_job, 548
describe_data_quality_job_definition, 771
describe_data_repository_associations, 415
describe_data_repository_tasks, 415
describe_data_set, 689
describe_data_set_permissions, 689
describe_data_set_refresh_properties, 689
describe_data_shares, 709
describe_data_shares_for_consumer, 709
describe_data_shares_for Producer, 709
describe_data_source, 489, 689
describe_data_source_permissions, 689
describe_data_sources, 554
describe_database, 894
describe_dataset, 219, 222, 408, 433, 548, 642, 721
describe_dataset_export_job, 642
describe_dataset_group, 408, 642
describe_dataset_import_job, 408, 642
describe_db_cluster_automated_backups, 698
describe_db_cluster_backtracks, 698
describe_db_cluster_endpoints, 587, 698
describe_db_cluster_parameter_groups, 298, 587, 698
describe_db_cluster_parameters, 298, 587, 698
describe_db_cluster_snapshot_attributes, 298, 587, 698
describe_db_cluster_snapshots, 298, 587, 698
describe_db_clusters, 298, 587, 698
describe_db_engine_versions, 298, 587, 698
describe_db_instance_automated_backups, 698
describe_db_instances, 298, 587, 698
describe_db_log_files, 698
describe_db_parameter_groups, 587, 698
describe_db_parameters, 587, 698
describe_db_proxies, 698
describe_db_proxy_endpoints, 698
describe_db_proxy_target_groups, 698
describe_db_proxy_targets, 698
describe_db_recommendations, 698
describe_db_security_groups, 698
describe_db_shard_groups, 698
describe_db_snapshot_attributes, 699
describe_db_snapshot_tenant_databases, 699
describe_db_snapshots, 699
describe_db_subnet_groups, 298, 587, 699
describe_default_cluster_parameters, 709
describe_default_parameters, 278
describe_deliveries, 152
describe_delivery_channel_status, 233
describe_delivery_channels, 233
describe_delivery_destinations, 152
describe_delivery_sources, 152
describe_delivery_stream, 397
describe_deployments, 619
describe_destinations, 152
describe_detector, 412
describe_device, 630, 771, 932
describe_device_fleet, 771
describe_device_job, 630
describe_device_policy_configuration, 932
describe_dhcp_options, 321
describe_dimension_keys, 650
describe_direct_connect_gateway_association_proposals, 288
describe_direct_connect_gateway_associations,
describe_direct_connect_gateway_attachments, 288
describe_direct_connect_gateways, 288
describe_directories, 292
describe_directory_configs, 54
describe_discoverer, 792
describe_document, 844
describe_document_classification_job, 222
describe_document_classifier, 222
describe_document_permission, 844
describe_document_versions, 930
describe_domain, 165, 611, 771, 880, 906, 932
describe_domain_auto_tunes, 360, 611
describe_domain_change_progress, 360, 611
describe_domain_config, 611
describe_domain_controllers, 292
describe_domain_endpoint_options, 127
describe_domain_health, 611
describe_domain_nodes, 611
describe_domains, 128, 611
describe_dominant_language_detection_job, 222
describe_draft_app_version_resources_import_status, 726
describe_drt_access, 831
describe_dry_run_progress, 611
describe_ecs_clusters, 619
describe_edge_deployment_plan, 771
describe_edge_packaging_job, 771
describe_effective_instance_associations, 844
describe_effective_patches_for_patch_baseline, 844
describe_effective_policy, 627
describe_egress_only_internet_gateways, 321
describe_eks_anywhere_subscription, 346
describe_elastic_gpus, 321
describe_elastic_ips, 619
describe_elastic_load_balancers, 619
describe_elasticsearch_domain, 360
describe_elasticsearch_domain_config, 360
describe_elasticsearch_domains, 360
describe_elasticsearch_instance_type_limits, 360
describe_email_monitoring_configuration, 936
describe_emergency_contact_settings, 831
describe_endpoint, 222, 383, 771
describe_endpoint_access, 709
describe_endpoint_authorization, 709
describe_endpoint_config, 771
describe_endpoint_group, 423
describe_endpoints, 307, 892, 894
describe_engagement, 848
describe_engine_default_cluster_parameters, 298, 587, 699
describe_engine_default_parameters, 350, 587, 699
describe_engine_versions, 575
describe_entities_detection_job, 222
describe_entities_detection_v2_job, 226
describe_entitlements, 54
describe_entity, 565, 936
describe_entity_aggregates, 440
describe_entity_aggregates_for_organization, 440
describe_entity_recognizer, 222
describe_environment_health, 354
describe_environment.managed_action_history, 354
describe_environment.managed_actions, 354
describe_environment.memories, 102, 104
describe_environment.resources, 354
describe_environment.status, 102, 104
describe_environments, 102, 104, 354
describe_evaluation_form, 238
describe_evaluations, 554
describe_event_aggregates, 440
describe_event_aggregates_for_organization, 440
describe_event_details, 440
describe_event_details_for_organization, 440
describe_event_source, 142, 383
describe_event_sources_config, 285
describe_event_subscriptions, 298, 587, 699, 709
describe_event_topics, 292
describe_event_tracker, 642
describe_event_types, 440
describe_events, 278, 298, 350, 354, 440, 575, 587, 623, 699, 709
describe_events_detection_job, 222
describe_events_for_organization, 440
describe_exclusions, 460
describe_execution, 828
describe_experience, 489
describe_experiment, 771
describe_explainability, 408
describe_explainability_export, 408
describe_export, 307, 522
describe_export_image_tasks, 321
describe_export_tasks, 152, 321, 699
describe_expressions, 128
describe_faq, 489
describe_fargate_profile, 346
describe_fast_launch_images, 321
describe_fast_snapshot_restores, 321
describe_feature_group, 771
describe_feature_metadata, 771
describe_feature_transformation, 642
describe_featured_results_set, 489
describe_feedback, 285
describe_fhir_datastore, 443
describe_fhir_export_job, 443
describe_fhir_import_job, 443
describe_file_caches, 415
describe_file_system_aliases, 415
describe_file_system_associations, 867
describe_file_system_policy, 343
describe_file_systems, 343, 415
describe_filter, 642
describe_findings, 460
describe_firewall, 594
describe_firewall_policy, 594
describe_fleet_history, 321
describe_fleet_instances, 321
describe_fleet_metadata, 932
describe_fleets, 54, 321
describe_flow_definition, 771
describe_flow_logs, 321
describe_flywheel, 222
describe_flywheel_iteration, 223
describe_folder, 689
describe_folder_contents, 930
describe_folder_permissions, 689
describe_folder_resolved_permissions, 689
describe_forecast, 408
describe_forecast_export_job, 408
describe_fpga_image_attribute, 321
describe_fpga_images, 321
describe_fraudster, 907
describe_fraudster_registration_job, 907
describe_function, 118
describe_gateway_information, 867
describe_gateway_route, 44
describe_generated_template, 113
describe_geofence_collection, 545
describe_global_clusters, 298, 587, 699
describe_global_networks, 598
describe_global_replication_groups, 350
describe_global_settings, 78
describe_global_table, 307
describe_global_table_settings, 307
describe_group, 454, 689, 936
describe_group_membership, 454, 689
describe_groups, 930
describe_handshake, 627
describe_hapg, 122
describe_health_service_status_for_organization, 440
describe_host_reservation_offers, 321
describe_host_reservations, 321
describe_hosted_connections, 288
describe_hosts, 321
describe_hours_of_operation, 238
describe_hsm, 122
describe_hsm_client_certificates, 709
describe_hsm_configurations, 709
describe_hub, 771, 800
describe_hub_content, 771
describe_human_loop, 68
describe_human_task_ui, 771
describe_hyper_parameter_tuning_job, 771
describe_model_bias_job_definition, 772
describe_model_card, 772
describe_model_card_export_job, 772
describe_model_explainability_job_definition, 772
describe_model_package, 772
describe_model_package_group, 772
describe_model_quality_job_definition, 772
describe_model_version, 548
describe_model_versions, 412
describe_monitor, 408
describe_monitoring_schedule, 772
describe_mount_target_security_groups, 343
describe_mount_targets, 343
describe_moving_addresses, 322
describe_multi_region_access_point_operation, 762
describe_my_user_profile, 619
describe_namespace, 689
describe_nat_gateways, 322
describe_network_acls, 322
describe_network_insights_access_scope_analyses, 322
describe_network_insights_access_scopes, 322
describe_network_insights_analyses, 322
describe_network_insights_paths, 322
describe_network_interface_attribute, 322
describe_network_interface_permissions, 322
describe_network_interfaces, 322
describe_nfs_file_shares, 867
describe_node, 630
describe_node_association_status, 623
describe_node_configuration_options, 709
describe_node_from_template_job, 630
describe_nodegroup, 346
describe_notebook_execution, 370
describe_notebook_instance, 772
describe_notebook_instance_lifecycle_config, 772
describe_notification_configurations, 772
describe_notification_rule, 208
describe_notification_subscriptions, 930
describe_notifications_for_budget, 101
describe_objects, 271
describe_obsvability_configuration, 50
describe_observation, 41
describe_operating_systems, 619
describe_ops_items, 844
describe_opt_out_lists, 664
describe_opted_out_numbers, 664
describe_option_group_options, 699
describe_option_groups, 699
describe_orderable_cluster_options, 709
describe_orderable_db_instance_options, 298, 587, 699
describe_organization, 627, 936
describe_organization_config_rule_statuses, 233
describe_organization_config_rules, 233
describe_organization_configuration, 282, 436, 464, 557, 800
describe_organization_conformance_pack_statuses, 233
describe_organization_conformance_packs, 233
describe_organization_health, 285
describe_organization_overview, 285
describe_organization_resource_collection_health, 285
describe_organizational_unit, 627
describe_organizations_access, 113
describe_outbound_connections, 611
describe_outbound_cross_cluster_search_connections, 360
describe_package, 165, 630
describe_package_group, 165
describe_package_import_job, 630
describe_package_version, 165, 630
describe_packages, 360, 611
describe_page, 848
describe_parameter_groups, 278, 575
describe_parameters, 278, 575, 844
describe_partner_event_source, 142, 383
describe_partners, 709
describe_patch_baselines, 844
describe_patch_group_state, 844
describe_patch_groups, 844
describe_patch_properties, 844
describe_pending_aggregation_requests, 233
describe_pending_maintenance_actions, 298, 587, 699
describe_permission_set, 860
describe_permission_set_provisioning_status, 860
describe_permissions, 619
describe_phi_detection_job, 226
describe_phone_number, 238
describe_phone_numbers, 664
describe_pii_entities_detection_job, 223
describe_pipe, 386
describe_pipeline, 772
describe_pipeline_definition_for_execution, 772
describe_pipeline_execution, 772
describe_PIPELINES, 271
describe_place_index, 545
describe_placement_groups, 322
describe_platform_version, 354
describe_pod_identity_association, 346
describe_policies, 71
describe_policy, 627
describe_pools, 665
describe_portfolio, 810
describe_portfolio_share_status, 810
describe_portfolio_shares, 810
describe_predefined_attribute, 238
describe_predictor, 408
describe_predictor_backtest_export_job, 408
describe_prefix_lists, 322
describe_principal_id_format, 322
describe_principal_mapping, 489
describe_problem, 41
describe_problem_observations, 41
describe_processing_job, 772
describe_product, 810
describe_product_as_admin, 810
describe_product_view, 810
describe_products, 800
describe_profiling_group, 188
describe_project, 201, 433, 772
describe_project_versions, 722
describe_projects, 722
describe_prompt, 238
describe_protection_configurations, 665
describe_protected_resource, 78
describe_protection, 831
describe_protection_group, 831
describe_provisioned_product, 810
describe_provisioned_product_plan, 810
describe_provisioning_artifact, 810
describe_provisioning_parameters, 810
describe_public_ipv_4_pools, 322
describe_publisher, 113
describe_publishing_destination, 436
describe_pull_request_events, 180
describe_pull_through_cache_rules, 334
describe_queries, 152
describe_query, 133
describe_query_definitions, 152
describe_query_suggestions_block_list, 489
describe_query_suggestions_config, 489
describe_queue, 238
describe_quick_connect, 238
describe_raid_arrays, 619
describe_rds_db_instances, 619
describe_receipt_rule, 820
describe_receipt_rule_set, 820
describe_recipe, 433, 642
describe_recommendation_export_jobs, 229
describe_recommendation_feedback, 191
describe_recommendation_service, 642
describe_record, 810
describe_recovery_instances, 304
describe_recovery_point, 78
describe_recovery_snapshots, 304
describe_redshift_idc_applications, 709
describe_refresh_schedule, 689
describe_region_settings, 78
describe_regions, 292, 322
describe_registration_attachments, 665
describe_registration_field_definitions, 665
describe_registration_field_values,
describe_scaling_activities, 35, 71
describe_scaling_parameters, 128
describe_scaling_plan_resources, 75
describe_scaling_plans, 75
describe_scaling_policies, 35
describe_scaling_process_types, 72
describe_schedule, 433
describe_scheduled_actions, 35, 72, 710
describe_scheduled_instance_availability, 322
describe_scheduled_instances, 322
describe_scheduled_query, 892
describe_scheduling_policies, 87
describe_schema, 642, 792
describe_scraper, 673
describe_secret, 795
describe_security_configuration, 370, 374
describe_security_group_references, 322
describe_security_group_rules, 322
describe_security_groups, 322
describe_security_profile, 238
describe_sender_ids, 665
describe_sentiment_detection_job, 223
describe_serverless_cache_snapshots, 351
describe_serverless_caches, 351
describe_servers, 623
describe_service, 50
describe_service_access_policies, 128
describe_service_action, 810
describe_service_action_execution_parameters, 810
describe_service_errors, 619
describe_service_integration, 285
describe_service_updates, 351, 575
describe_services, 340, 670, 874
describe_sessions, 54, 844
describe_settings, 292
describe_severity_levels, 875
describe_shared_directories, 292
describe_shared_vpc_configuration, 415
describe_slot, 522
describe_slot_type, 522
describe_smb_file_shares, 867
describe_smb_settings, 868
describe_snapshot_attribute, 322
describe_snapshot_copy_grants, 710
describe_snapshot_schedule, 868
describe_snapshot_schedules, 710
describe_snapshot_tier_status, 322, 351, 415, 575
describe_snomedct_inference_job, 226
describe_solution, 642
describe_solution_version, 642
describe_source_networks, 304
describe_source_regions, 699
describe_source_servers, 304
describe_space, 772
describe_spacer, 907
describe_speaker_enrollment_job, 907
describe_spend_limits, 665
describe_spot_datafeed_subscription, 322
describe_spot_fleet_instances, 322
describe_spot_fleet_request_history, 322
describe_spot_fleet_requests, 322
describe_spot_instance_requests, 322
describe_spot_price_history, 322
describe_ssl_policies, 367
describe_stack_drift_detection_status, 113
describe_stack_events, 113
describe_stack_instance, 113
describe_stack_provisioning_parameters, 619
describe_stack_resource, 113
describe_stack_resource_drifts, 113
describe_stack_resources, 113
describe_stack_set, 114
describe_stack_set_operation, 114
describe_stack_summary, 619
describe_stacks, 54, 114, 619
describe_stale_security_groups, 322
describe_standards, 800
describe_standards_controls, 800
describe_state_machine, 828
describe_state_machine_alias, 828
describe_state_machine_for_execution, 828
describe_statement, 713
describe_step, 370
describe_storage, 710
describe_storage_virtual_machines, 415
describe_store_image_tasks, 322
describe_storedi_scsi_volumes, 868
describe_stream, 311, 497
describe_stream_consumer, 497
describe_stream_processor, 722
describe_stream_summary, 497
describe_studio, 370
describe_studio_lifecycle_config, 772
describe_subnet_groups, 278, 575
describe_subnets, 322
describe_subscribed_workteam, 772
describe_subscribers_for_notification, 101
describe_subscription, 831
describe_subscription_filters, 153
describe_suggesters, 128
describe_supported_languages, 875
describe_table, 307, 713, 894
describe_table_replica_auto_scaling, 307
describe_table_restore_status, 710
describe_tag_option, 811
describe_tags, 72, 289, 322, 343, 363, 367, 554, 710, 943
describe_tape_archives, 868
describe_tape_recovery_points, 868
describe_tapes, 868
describe_target_group_attributes, 367
describe_target_groups, 367
describe_target_health, 367
describe_targeted_sentiment_detection_job, 223
describe_task_definition, 340
describe_task_sets, 340
describe_tasks, 340
describe_template, 689
describe_template_alias, 689
describe_template_definition, 689
describe_template_permissions, 689
describe_tenant_databases, 699
describe_termination_policy_types, 72
describe_test_cases, 168
describe_test_execution, 522
describe_test_set, 522
describe_test_set_discrepancy_report, 522
describe_test_set_generation, 522
describe_text_translation_job, 900
describe_theme, 689
describe_theme_alias, 689
describe_theme_permissions, 689
describe_thesaurus, 489
describe_time_based_auto_scaling, 619
describe_time_to_live, 307
describe_tls_inspection_configuration, 594
describe_topic, 689
describe_topic_permissions, 689
describe_topic_refresh, 689
describe_topic_refresh_schedule, 689
describe_topics_detection_job, 223
describe_tracker, 545
describe_traffic_distribution_group, 238
describe_traffic_mirror_filters, 322
describe_traffic_mirror_sessions, 322
describe_traffic_mirror_targets, 322
describe_traffic_sources, 72
describe_trails, 133
describe_training_job, 772
describe_transaction, 511
describe_transform_job, 772
describe_transit_gateway_attachments, 322
describe_transit_gateway_connect_peers, 323
describe_transit_gateway_connects, 323
describe_transit_gateway_multicast_domains, 323
describe_transit_gateway_peering_attachments, 323
describe_transit_gateway_policy_tables, 323
describe_transit_gateway_route_table_announcements, 323
describe_transit_gateway_route_tables, 323
describe_transit_gateway_vpc_attachments, 323
describe_transit_gateways, 323
describe_trial, 772
describe_trial_component, 772
describe_trunk_interface_associations, 323
describe_trust_store_associations, 367
describe_trust_store_revocations, 367
describe_trust_stores, 367
describe_trusted_advisor_check_refresh_status, 875
describe_trusted_advisor_check_result, 875
describe_trusted_advisor_check_summaries, 875
describe_trusted_advisor_checks, 875
describe_trusted_token_issuer, 860
describe_trusts, 292
describe_type, 114
describe_type_registration, 114
describe_update, 346
describe_update_actions, 351
describe_update_directory, 292
describe_upload_buffer, 868
describe_usage_limits, 710
describe_usage_report_subscriptions, 54
describe_user, 238, 454, 578, 689, 936
describe_user_groups, 351
describe_user_hierarchy_group, 238
describe_user_hierarchy_structure, 238
describe_user_import_job, 215
describe_user_pool, 215
describe_user_pool_client, 215
describe_user_pool_domain, 215
describe_user_profile, 201, 772
describe_user_profiles, 619
describe_user_stackAssociations, 54
describe_users, 54, 351, 575, 930
describe_valid_db_instance_modifications, 587, 699
describe_vault, 419
describe_verified_access_endpoints, 323
describe_verified_access_groups, 323
describe_verified_access_instance_Logging_configurations, 323
describe_verified_access_instances, 323
describe_verified_access_trust_providers, 323
describe_verified_destination_numbers, 665
describe_view, 238, 252
describe_virtual_cluster, 374
describe_virtual_gateway, 44
describe_virtual_gateways, 289
describe_virtual_interfaces, 289
describe_virtual_node, 44
describe_virtual_router, 44
describe_virtual_service, 44
describe_vocabulary, 238
describe_voices, 668
describe_volume_attribute, 323
describe_volume_status, 323
describe_volumes, 323, 415, 619
describe_volumes_modifications, 323
describe_vpc_attribute, 238
describe_vpc_classic_link, 323
describe_vpc_classic_link_dns_support, 323
describe_vpc_connection, 483, 689
describe_vpc_connector, 50
describe_vpc_endpoint_connection_notifications, 323
describe_vpc_endpoint_connections, 323
describe_vpc_endpoint_service_configurations, 323
describe_vpc_endpoint_service_permissions, 323
describe_vpc_endpoint_services, 323
describe_vpc_endpoints, 323, 360, 611
describe_vpc_ingress_connection, 50
describe_vpc_peering_connections, 323
describe_vpcs, 323
describe_vpn_connections, 323
describe_vpn_gateways, 323
describe_vtl_devices, 868
describe_warm_pool, 72
describe_worklist, 907
describe_website_certificate_authority, 54
describe_what_if_analysis, 408
describe_what_if_forecast, 408
describe_what_if_forecast_export, 408
describe_worker_configuration, 486
describe_workflow_execution, 880
describe_workflow_type, 880
describe_workforce, 772
describe_working_storage, 868
describe_workspace, 561, 673
describe_workspace_associations, 943
describe_workspace_authentication, 561
describe_workspace_bundles, 943
describe_workspace_configuration, 561
describe_workspace_directories, 943
describe_workspace_image_permissions, 943
describe_workspace_images, 943
describe_workspace_snapshots, 943
describe_workspaces, 943
describe_workspaces_connection_status, 943
describe_workteam, 772
detach_certificate_from_distribution, 540
detach_classic_link_vpc, 323
detach_customer_managed_policy_reference_from_permission_set, 860
detach_disk, 540
detach_elastic_load_balancer, 619
detach_from_index, 109
detach_group_policy, 446
detach_instances, 72
detach_instances_from_load_balancer, 540
detach_internet_gateway, 323
detach_load_balancer_from_subnets, 364
detach_load_balancer_target_groups, 72
detach_load_balancers, 72
detach_managed_policy_from_permission_set, 860
detach_network_interface, 323
detach_object, 109
detach_policy, 109, 627
detach_role_policy, 446
detach_static_ip, 540
detach_traffic_sources, 72
detach_typed_link, 109
detach_user_policy, 446
detach_verified_access_trust_provider, 323
detach_volume, 323, 868
detach_vpn_gateway, 323
detect_add_on, 540
detect_address_transfer, 323
detect_alarm_actions, 139
detect_application_layer_automatic_response, 831
detect_availability_zones_for_load_balancer, 364
detect_aws_network_performance_metric_subscription, 323
detect_aws_service_access, 811
detect_aws_service_access, 627
detectBaseline, 258
detect_client_authentication, 292
detect_control, 258
detect_crl, 451
detect_delegated_admin_account, 464
detect_directory, 109
detect_domain_auto_renew, 741
detect_domain_transfer_lock, 741
detect_ebs_encryption_by_default, 323
detect_enhanced_monitoring, 497
detect_fast_launch, 323
detect_fast_snapshot_restores, 323
detect_federation, 133
detect_gateway, 868
detect_health_service_access_for_organization, 440
disable_hosted_zone_dnssec, 738
disable_http_endpoint, 699
disable_image, 323
disable_image_block_public_access, 323
disable_image_deprecation, 323
disable_image_deregistration_protection, 323
disable_import_findings_for_product, 800
disable_insight_rules, 139
disable_ipam_organization_admin_account, 323
disable_key, 507
disable_key_rotation, 507
disable_kinesis_streaming_destination, 307
disable_ldaps, 292
disable_logging, 710
disable_macie, 557
disable_metrics_collection, 72
disable_organization_admin_account, 282, 436, 557, 800
disable_policy_type, 627
disable_proactive_engagement, 831
disable_profile, 451
disable_radius, 292
disable_region, 13
disable_rule, 142, 383
disable_sagemaker_servicecatalog_portfolio, 772
disable_security_hub, 800
disable_serial_console_access, 324
disable_snapshot_block_public_access, 324
disable_snapshot_copy, 710
disable_sso, 292
disable_stage_transition, 198
disable_transit_gateway_route_table_propagation, 324
disable_trust_anchor, 451
disable_user, 54, 394
disable_vgw_route_propagation, 324
disable_vpc_classic_link, 324
disable_vpc_classic_link_dns_support, 324
disassociate_access_policy, 346
disassociate_accounts, 93
disassociate_address, 324
disassociate_admin_account, 402
disassociate_analytics_data_set, 238
disassociate_app_block_builder_app_block, 54
disassociate_application_fleet, 54
disassociate_application_from_entitlement, 54
disassociate_approval_rule_template_from_repository, 180
disassociate_approved_origin, 238
disassociate_assessment_report_evidence_folder, 65
disassociate_attribute_group, 46
disassociate_bot, 238
disassociate_browser_settings, 946
disassociate_budget_from_resource, 811
disassociate_client_vpn_target_network, 324
disassociate_connect_peer, 598
disassociate_connection_alias, 943
disassociate_connection_from_lag, 289
disassociate_custom_domain, 50
disassociate_customer_gateway, 598
disassociate_data_share_consumer, 710
disassociate_default_view, 729
disassociate_delegate_from_resource, 946
disassociate_delegation_signer_from_domain, 741
disassociate_domain, 932
disassociate_drt_log_bucket, 831
disassociate_drt_role, 831
disassociate_elastic_ip, 619
disassociate_enclave_certificate_iam_role, 324
disassociate_entitites_from_experience, 489
disassociate_environment_operations_role, 354
disassociate_external_connection, 165
disassociate_faces, 722
disassociate_file_system, 868
disassociate_file_system_aliases, 415
disassociate_firewall_rule_group, 753
disassociate_fleet, 54
disassociate_flow, 238
disassociate_fraudster, 907
disassociate_from_administrator_account,
INDEX

disassociate_from_master_account, 436, 557, 800

disassociate_gateway_from_server, 82

disassociate_global_replication_group, 351

disassociate_health_check, 831

disassociate_iam_instance_profile, 324

disassociate_identity_provider_config, 347

disassociate_instance_event_window, 324

disassociate_instance_storage_config, 238

disassociate_ip_access_settings, 946

disassociate_ip_groups, 943

disassociate_ipam_byoasn, 324

disassociate_ipam_resource_discovery, 324

disassociate_kms_key, 153

disassociate_lambda_function, 238

disassociate_lenses, 925

disassociate_lex_bot, 238

disassociate_license, 561

disassociate_link, 598

disassociate_mac_sec_key, 289

disassociate_member, 464, 557

disassociate_member_from_group, 936

disassociate_members, 436, 800

disassociate_membership, 282

disassociate_nat_gateway_address, 324

disassociate_network_settings, 946

disassociate_node, 623

disassociate_ops_item_related_item, 844

disassociate_origination_identity, 665

disassociate_personas_from_entities, 489

disassociate_phone_number_contact_flow, 238

disassociate_pricing_rules, 95

disassociate_principal_from_portfolio, 811

disassociate_product_from_portfolio, 811

disassociate_profiles, 925

disassociate_protect_configuration, 665

disassociate_qualification_from_worker, 580

disassociate_queue_quick_connects, 238

disassociate_recovery_point, 78

disassociate_recovery_point_from_parent, 78

disassociate_repository, 191

disassociate_resolver_endpoint_ip_address, 753

disassociate_resolver_query_log_config, 754

disassociate_resolver_rule, 754

disassociate_resource, 46, 883

disassociate_resource_share, 694

disassociate_resource_share_permission, 694

disassociate_route_table, 324

disassociate_routing_profile_queues, 238

disassociate_security_key, 238

disassociate_service_action_from_provisioning_artifact, 811

disassociate_service_quota_template, 817

disassociate_subnet_cidr_block, 324

disassociate_subnets, 594

disassociate_tag_option_from_resource, 811

disassociate_team_member, 201

disassociate_third_party_firewall, 402

disassociate_tracker_consumer, 545

disassociate_traffic_distribution_group_user, 238

disassociate_transit_gateway_connect_peer, 598

disassociate_transit_gateway_multicast_domain, 324

disassociate_transit_gateway_policy_table, 324

disassociate_transit_gateway_route_table, 324

disassociate_trial_component, 772

disassociate_trunk_interface, 324

disassociate_trust_store, 946

disassociate_user, 536

disassociate_user_access_logging_settings, 946

disassociate_user_from_permission_group,
disassociate_user_proficiencies, 238
disassociate_user_settings, 946
disassociate_vpc_cidr_block, 324
disassociate_vpc_from_hosted_zone, 738
disassociate_web_acl, 917, 922
disassociate_website_authorization_provider, 933
disassociate_website_certificate_authority, 933
disassociate_website_application, 943
discard_registration_version, 665
disconnect_custom_key_store, 507
disconnect_participant, 252, 480
disconnect_recoveryInstance, 304
disconnect_source_server, 304
disconnect_user, 475
discover_input_schema, 500, 503
discover_instances, 814
discover_instances_revision, 814
discover_poll_endpoint, 340
dismiss_user_contact, 238
dispose_package_versions, 165
dissociate_access_grants_identity_center, 762
dissociate_package, 360, 611
distribute_dataset_entries, 722
dl, 293
docdb, 295
docdbelastic, 299
domain_metadata, 834
download_db_log_file_portion, 699
download_default_key_pair, 540
download_file, 757
drs, 301
dynamodb, 305
dynamodbstreams, 309
ebs, 311
e2, 314
e2instanceconnect, 329
ecr, 331
ecrpublic, 335
ecs, 337
efs, 341
eks, 344
elasticache, 348
elasticbeanstalk, 352
enable_key_rotation, 507
enable_kinesis_streaming_destination, 307
enable_ldaps, 292
enable_logging, 710
enable_mace, 557
enable_metrics_collection, 72
enable_mfa_device, 446
enable_organization_admin_account, 282, 436, 557, 800
enable_policy_type, 627
enable_proactive_engagement, 831
enable_profile, 451
enable_radius, 292
enable_reachability_analyzer_organization_sharing, 324
enable_region, 13
enable_rule, 142, 383
enable_sagemaker_servicecatalog_portfolio, 772
enable_security_hub, 800
enable_serial_console_access, 324
enable_sharing_with_aws_organization, 694
enable_snapshot_block_public_access, 324
enable_snapshot_copy, 710
enable_sso, 292
enable_stage_transition, 198
enable_transit_gateway_route_table_propagation, 324
enable_trust_anchor, 451
enable_user, 54, 394
enable_vgw_route_propagation, 324
enable_volume_io, 324
enable_vpc_classic_link, 324
enable_vpc_classic_link_dns_support, 324
encrypt, 507
encrypt_data, 636
enter_standby, 72
entityresolution, 377
estimate_template_cost, 114
evaluate_expression, 271
evaluate_feature, 145
evaluate_pull_request_approval_rules, 180
evaluate_session, 907
eventbridge, 380
eventbridgepipes, 384
eventbridgescheduler, 386
execute_budget_action, 101
execute_change_set, 114
execute_command, 340
execute_core_network_change_set, 598
execute_fast_reset, 590
execute_gremlin_explain_query, 590
execute_gremlin_profile_query, 590
execute_gremlin_query, 590
execute_open_cypher_explain_query, 590
execute_open_cypher_query, 590
execute_policy, 72
execute_provisioned_product_plan, 811
execute_provisioned_product_service_action, 811
execute_scheduled_query, 892
execute_sql, 703
execute_statement, 307, 703, 713
execute_transaction, 307
expire_session, 54
export_api, 28
export_auto_scaling_group_recommendations, 229
export_backup_plan_template, 78
export_certificate, 15
export_client_vpn_client_certificate_revocation_list, 324
export_client_vpn_client_configuration, 324
export_client_vpn_client_workbench poundedformaton, 324
export_earth_observation_job, 782
export_ebs_volume_recommendations, 229
export_ec2_instance_recommendations, 229
export_ecs_service_recommendations, 229
export_image, 324
export_journal_to_s3, 682
export_key, 633
export_lambda_function_recommendations, 229
export_lens, 925
export_license_recommendations, 229
export_notebook, 60
export_schema, 792
export_server_engine_attribute, 623
export_serverless_cache_snapshot, 351
export_snapshot, 540
export_source_network_cfn_template, 304
export_table_to_point_in_time, 307
export_transit_gateway_routes, 324
export_vector_enrichment_job, 782
extend_license_consumption, 531
extend_transaction, 511
failover_db_cluster, 298, 587, 699
failover_global_cluster, 587, 699
failover_global_replication_group, 351
failover_primary_compute, 710
failover_shard, 575
filter_log_events, 153
finspace, 389
finspacedata, 392
firehose, 395
fis, 397
flush_stage_authorizers_cache, 21
flush_stage_cache, 21
fms, 400
forecastqueryservice, 403
forecastservice, 405
forget_device, 215
forgot_password, 215
frauddetector, 409
fsx, 413
generate_bot_element, 522
generate_card_validation_data, 636
generate_client_certificate, 21
generate_credential_report, 446
generate_data_key, 507
generate_data_key_pair, 507
generate_data_key_pair_without_plaintext, 507
generate_data_key_without_plaintext, 507
generate_data_set, 567
generate_embed_url_for_anonymous_user, 689
generate_embed_url_for_registered_user, 689
generate_mac, 507, 636
generate_mobile_sdk_release_url, 922
generate_organizations_access_report, 446
generate_pin_data, 636
generate_presigned_url, 757
generate_random, 507
generate_service_last_accessed_details, 446
generate_service_last_accessed_details, 446
generate_service_last_accessed_details, 446
generate_service_last_accessed_details, 446
get_access_control_effect, 936
get_access_grant, 762
get_access_grants_instance, 762
get_access_grants_instance_for_prefix, 762
get_access_grants_instance_resource_policy, 762
get_access_grants_location, 762
get_access_key_info, 871
get_access_key_last_used, 446
get_access_log_subscription, 910
get_access_point, 762
get_access_point_configuration_for_object_lambda, 762
get_access_point_for_object_lambda, 762
get_access_point_policy, 762
get_access_point_policy_for_object_lambda, 762
get_access_point_policy_status, 762
get_access_point_policy_status_for_object_lambda, 762
get_access_policy, 615
get_access_preview, 10
get_access_token, 531
get_account, 21, 658, 824
get_account_alias, 878
get_account_authorization_details, 446
get_account_balance, 580
get_account_configuration, 15, 193
get_account_level_service_configuration, 729
get_account_limit, 738
get_account_link, 943
get_account_password_policy, 446
get_account_sending_enabled, 820
get_account_settings, 515, 615, 678, 732
get_account_status, 65
get_account_summary, 446
get_accuracy_metrics, 408
get_action, 399
get_action_recommendations, 647
get_action_type, 198
get_active_names, 540
get_activity_task, 828
get_adapter, 889
get_adapter_version, 889
get_adm_channel, 653
get_admin_account, 402
get_admin_scope, 402
get_administrator_account, 436, 557, 800
get_aggregate_compliance_details_by_config_rule, 233
get_aggregate_config_rule_compliance_summary, 233
get_aggregate_conformance_pack_compliance_summary, 233
get_aggregate_discovered_resource_counts, 233
get_aggregate_resource_config, 233
get_alarms, 540
get_alias, 313, 633
get_allow_list, 557
get_alternate_contact, 13
get_analyzed_resource, 10
get_analyzer, 10
get_annotation_import_job, 605
get_annotation_store, 605
get_annotation_store_version, 605
get_anomalies, 264
get_anomaly_group, 551
get_anomaly_monitors, 264
get_anomaly_subscriptions, 264
get_answer, 925
get_api, 28
get_api_key, 21
get_api_keys, 21
get_api_mapping, 28
get_api_mappings, 28
get_apps, 653
get_apps_list, 402
get_application_assignment_configuration, 860
get_application_authentication_method, 860
get_application_date_range_kpi, 653
get_application_grant, 860
get_application_policy, 807
get_application_revision, 185
get_application_settings, 653
get_applied_schema_version, 110
get_approximate_usage_records, 264
get_apps, 653
get_attachment, 252
get_attribute_group, 47
get_attribute_values, 670
get_attributes, 834
get_auth_policy, 910
get_authorization_token, 165, 334, 337
get_authorizer, 21, 28
get_authorizers, 22, 28
get_auto_merging_preview, 267
get_auto_scaling_group_recommendations, 229
get_auto_snapshots, 540
get_auto_termination_policy, 370
get_automated_discovery_configuration, 557
get_authorization_token, 165, 334, 337
get_authorizer, 21, 28
get_authorizers, 22, 28
get_auto_merging_preview, 267
get_auto_scaling_group_recommendations, 229
get_auto_snapshots, 540
get_auto_termination_policy, 370
get_automated_discovery_configuration, 557
get_authorization_token, 165, 334, 337
get_authorizer, 21, 28
get_authorizers, 22, 28
get_auto_merging_preview, 267
get_auto_scaling_group_recommendations, 229
get_auto_snapshots, 540
get_auto_termination_policy, 370
get_automated_discovery_configuration, 557
get_authorization_token, 165, 334, 337
get_authorizer, 21, 28
get_authorizers, 22, 28
get_auto_merging_preview, 267
get_auto_scaling_group_recommendations, 229
get_auto_snapshots, 540
get_auto_termination_policy, 370
get_automated_discovery_configuration, 557
INDEX

get_aws_default_service_quota, 817
get_aws_network_performance_data, 324
get_aws Organizations_access_status, 811
get_backup_plan, 78
get_backup_plan_from_json, 78
get_backup_plan_from_template, 78
get_backup_selection, 78
get_backup_vault_access_policy, 78
get_backup_vault_notifications, 78
get_baidu_channel, 653
get_bandwidth_rate_limit_schedule, 82
get_base_path_mapping, 22
get_base_path_mappings, 22
get_baseline, 258
get_baseline_operation, 258
get_batch_import_jobs, 412
get_batch_prediction, 554
get_batch_prediction_jobs, 412
get_billing_group_cost_report, 95
get_blacklist_reports, 658, 824
get_blob, 180
get_block_encryption, 757
get_block_public_access_configuration, 370
get_blueprint, 427
get_blueprint_run, 427
get_blueprint_runs, 427
get_blueprints, 540
get_bootstrap_brokers, 483
get_bot, 518
get_bot_alias, 518
get_bot_aliases, 518
get_bot_channel_association, 518
get_bot_channel_associations, 518
get_bot_versions, 518
get_bots, 518
get_branch, 180
get_browser_settings, 946
get_bucket, 762
get_bucket_accelerate_configuration, 757
get_bucket_access_keys, 540
get_bucket_acl, 757
get_bucket_analytics_configuration, 757
get_bucket_bundles, 540
get_bucket_cors, 757
get_bucket_encryption, 757
get_bucket_intelligent_tiering_configuration, 757
get_bucket_inventory_configuration, 757
get_bucket_lifecycle, 757
get_bucket_lifecycle_configuration, 757, 762
get_bucket_location, 757
get_bucket_logging, 757
get_bucket_metrics_data, 540
get_bucket_metrics_configuration, 757
get_bucket_notification, 757
get_bucket_notification_configuration, 757
get_bucket_ownership_controls, 757
get_bucket_policy, 757, 762
get_bucket_policy_status, 757
get_bucket_replication, 757, 762
get_bucket_request_payment, 757
get_bucket_statistics, 557
get_bucket_tagging, 757, 762
get_bucket_versioning, 757, 762
get_bucket_website, 758
get_buckets, 540
get builtin_intent, 518
get builtin_intents, 519
get bulk_publish_details, 219
get bundles, 540
get_byte_match_set, 913, 917
get_cache_policy, 118
get_cache_policy_config, 118
get calculated_attribute_for_profile, 267
get calculated_attribute_definition, 267
get calculation_execution, 60
get calculation_execution_code, 60
get calculation_execution_status, 60
get calendar_state, 844
get call_analytics_category, 897
get call_analytics_job, 897
get caller_identity, 871
get campaign, 653
get campaign activities, 653
get campaign_date_range_kpi, 653
get campaign_state, 244
get_campaign_state_batch, 244
get_campaign_version, 653
get_campaign_versions, 633
get_campaigns, 653
get_canary, 883
get_canary_runs, 883
get_capacity_assignment_configuration, 60
get_capacity_reservation, 60
get_capacity_reservation_usage, 324
get_case, 246
get_case_audit_events, 246
get_case_event_configuration, 247
get_catalog_import_status, 427
get_celebrity_info, 722
get_celebrity_recognition, 722
get_cell, 750
get_cell_readiness_summary, 750
get_certificate, 15, 18
get_certificate_authority_certificate, 18
get_certificate_authority_csr, 18
get_certificates, 540
get_change, 738
get_change_logs, 65
get_change_token, 913, 917
get_change_token_status, 913, 917
get_changeset, 394
get_channel, 133, 470
get_channels, 653
get_checker_ip_ranges, 738
get_chunk, 84
get_cis_scan_report, 464
get_cis_scan_result_details, 464
get_classification_export_configuration, 557
get_classification_scope, 557
get_classifier, 427
get_classifiers, 427
get_client_certificate, 22
get_client_certificates, 22
get_cloud_formation_stack_records, 540
get_cloud_formation_template, 807
get_cloud_front_origin_access_identity, 118
get_cloud_front_origin_access_identity_configget
get_cluster_credentials, 710
get_cluster_credentials_with_iam, 710
get_cluster_policy, 483
get_cluster_session_credentials, 370
get_cluster_snapshot, 301
get_code_binding_source, 792
get_code_signing_config, 515
get_cognito_events, 219
get_coip_pool_usage, 324
get_column_statistics_for_partition, 427
get_column_statistics_for_table, 427
get_column_statistics_task_run, 427
get_column_statistics_task_runs, 427
get_command_invocation, 844
get_comment, 180
get_comment_reactions, 180
get_comments_for_compared_commit, 180
get_comments_for_pull_request, 180
get_commit, 180
get_compatible_elasticsearch_versions, 360
get_compatible_kafka_versions, 483
get_compatible_versions, 611
get_compliance_detail, 402
get_compliance_details_by_config_rule, 233
get_compliance_details_by_resource, 233
get_compliance_summary, 735
get_compliance_summary_by_config_rule, 233
get_compliance_summary_by_resource_type, 233
get_component, 457, 678, 854
get_component_policy, 457
get_composition, 480
get_config, 122
get_configuration, 47, 464
get_configuration_policy, 800
get_configuration_policy_association, 800
get_configuration_set, 658, 824
get_configuration_set_event_destinations, 658, 661, 824
get_conformance_pack_compliance_details, 233
get_conformance_pack_compliance_summary,
get_connect_attachment, 598
get_connect_instance_config, 244
get_connect_peer, 598
get_connect_peer_associations, 598
get_connection, 25, 205, 427
get_connection_status, 844
get_connections, 427, 598
get_connector, 639
get_console_output, 325
get_console_screenshot, 325
get_consolidated_report, 925
get_contact, 824, 848
get_contact_attributes, 238
get_contact_channel, 848
get_contact_information, 13
get_contact_list, 824
get_contact_methods, 540
get_contact_policy, 848
get_contact_reachability_status, 741
get_container_api_metadata, 540
get_container_images, 540
get_container_log, 540
get_container_recipe, 457
get_container_recipe_policy, 457
get_container_service_deployments, 540
get_container_service_metric_data, 540
get_container_service_powers, 540
get_container_services, 540
get_content, 254
get_content_moderation, 722
get_content_summary, 254
get_context_keys_for_custom_policy, 446
get_context_keys_for_principal_policy, 446
get_continuous_deployment_policy, 118
get_continuous_deployment_policy_config, 118
get_control, 65
get_control_operation, 258
get_core_network, 598
get_core_network_change_events, 598
get_core_network_change_set, 598
get_core_network_policy, 598
get_cost_and_usage, 264
get_cost_and_usage_with_resources, 264
get_cost_categories, 264
get_cost_estimate, 540
get_cost_estimation, 285
get_cost_forecast, 264
get_coverage_statistics, 436
get_crawler, 427
get_crawler_metrics, 427
get_crawlers, 427
get_credential_report, 446
get_credentials, 716
get_credentials_for_identity, 211
crl, 451
get_csv_header, 216
get_current_metric_data, 238
get_current_user, 930
get_current_user_data, 238
generate_salesforce_custom_domain, 557
generate_salesforce_custom_domain_association, 716
generate_salesforce_custom_entity_type, 427
generate_salesforce_custom_model, 90
generate_salesforce_custom_rule_policy, 233
generate_salesforce_custom_verification_email_template, 820, 824
generate_salesforce_customer_gateway_associations, 598
generate_salesforce_dashboard, 139
generate_salesforce_dashboard_embed_url, 689
generate_salesforce_dashboard_for_job_run, 376
generate_salesforce_data_access, 762
generate_salesforce_data_catalog, 60
generate_salesforce_data_catalog_encryption_settings, 427
generate_salesforce_data_cells_filter, 511
generate_salesforce_data_lake_exception_subscription, 804
generate_salesforce_data_lake_organization_configuration, 804
generate_salesforce_data_lake_settings, 511
generate_salesforce_data_lake_sources, 804
generate_salesforce_data_protection_policy, 153, 837
generate_salesforce_data_quality_metrics, 552
generate_salesforce_data_quality_result, 427
generate_salesforce_data_quality_rule_recommendation_run, 427
generate_salesforce_data_quality_ruleset, 427
generate_salesforce_data_quality_ruleset_evaluation_run, 427
generate_salesforce_data_retrieval_policy, 419
generate_salesforce_data_source, 274, 555, 611
generate_salesforce_data_source_run, 274
Index

get_data_view, 394
get_database, 60, 427, 854
get_databases, 427
get_dataset, 394
get_decrypted_api_key, 922
get_dedicated_ip, 658, 824
get_dedicated_ip_pool, 824
get_dedicated_ips, 658, 824
get_default_credit_specification, 325
get_default_patch_baseline, 844
get_default_retention_policy, 936
get_default_scraper_configuration, 673
get_default_view, 729
get_delegated_admin_account, 464
get_delegations, 65
get_delete_events_by_event_type_status, 412
get_deliverability_dashboard_options, 658, 824
get_deliverability_test_report, 658, 824
get_delivery, 153
get_delivery_destination, 153
get_delivery_destination_policy, 153
get_delivery_source, 153
get_deployable_patch_snapshot_for_instance, 844
get_deployment, 22, 28, 185, 678
get_deployment_config, 185
get_deployment_group, 185
get_deployment_instance, 185
get_deployment_target, 185
get_deployments, 22, 28, 777
get_detector, 436
get_detector_version, 412
get_detectors, 412
get_dev_endpoint, 427
get_dev_endpoints, 427
get_dev_environment, 173
get_device, 98, 216
get_device_fleet_report, 772
get_device_position, 545
get_device_position_history, 545
get_device_registration, 777
get_devices, 598
get_differences, 180
get_digest, 682
get_dimension_key_details, 650
get_dimension_values, 264
get_directory, 110
get_directory_limits, 292
get_directory_registration, 639
get_discovered_resource_counts, 233
get_discovered_schema, 792
get_disk, 541
get_disk_snapshot, 541
get_disk_snapshots, 541
get_disks, 541
generate_distribution, 118
generate_distribution_bundles, 541
generate_distribution_config, 118
generate_distribution_configuration, 457
generate_distribution_latest_cache_reset, 541
generate_distribution_metric_data, 541
generate_distributions, 541
generate_dnssec, 738
generate_document, 844, 930
generate_document_analysis, 889
generate_document_path, 930
generate_document_text_detection, 889
generate_document_version, 930
generate_documentation_part, 22
generate_documentation_parts, 22
generate_documentation_version, 22
generate_documentation_versions, 22
generate_domain, 247, 267, 274, 541
generate_domain_deliverability_campaign, 658, 824
generate_domain_detail, 741
generate_domain_maintenance_status, 611
generate_domain_name, 22, 28
generate_domain_names, 22, 28
generate_domain_permissions_policy, 165
generate_domain_statistics_report, 658, 824
generate_domain_suggestions, 741
generate_domains, 541
generate_download_url_for_layer, 334
generate_earth_observation_job, 782
generate_ebs_default_kms_key_id, 325
generate_ebs_encryption_by_default, 325
generate_ebs_volume_recommendations, 229
generate_ec2_instance_recommendations, 229
generate_ec2_recommendation_projected_metrics, 229
get_ec_2_deep_inspection_configuration,
464
get_ecs_service_recommendation_projected_metrics
229
get_ecs_service_recommendations, 229
get_effective_permissions_for_path,
511
get_effective_recommendation_preferences,
229
get_email_channel, 653
get_email_identity, 658, 824
get_email_identity_policies, 824
get_email_template, 653, 824
get_enabled_baseline, 258
get_enabled_control, 258
get_enabled_standards, 800
get_encoder_configuration, 480
get_encryption_config, 949
get_encryption_key, 464
get_endpoint, 653
get_endpoint_access, 716
get_endpoint_attributes, 837
get_engine_status, 590
get_enrollment_status, 229
get_enrollment_statuses_for_organization,
229
get_entitlements, 569
get_entity_types, 412
get_environment, 274, 391, 584, 678
get_environment_account_connection,
678
get_environment_blueprint, 274
get_environment_blueprint_configuration,
274
get_environment_profile, 274
get_environment_template, 678
get_environment_template_version, 678
gettle_u, 601
get_evaluation, 555
get_evaluation_job, 90
get_event, 412
gcter_event_data_store, 133
gcter_event_prediction, 412
gcter_event_prediction_metadata, 412
gcter_event_selectors, 133
gcter_event_source_mapping, 515
gcter_event_stream, 267, 653
gcter_event_types, 412
gcter_evidence, 65
gcter_evidence_by_evidence_folder, 65
gcter_evidence_file_upload_url, 65
gcter_evidence_folder, 65
gcter_evidence_folders_by_assessment, 65
gcter_evidence_folders_by_assessment_control, 65
gcter_exclusions_preview, 461
gcter_execution_history, 828
gcter_expense_analysis, 889
gcter_experiment, 145, 399
gcter_experiment_results, 145
gcter_experiment_target_account_configuration,
399
gcter_experiment_template, 399
gcter_export, 22, 519
gcter_export_job, 654, 824
gcter_export_jobs, 654
gcter_export_snapshot_records, 541
gcter_external_data_view_access_details,
394
gcter_external_models, 412
gcter_face_detection, 722
gcter_face_liveness_session_results, 722
gcter_face_search, 722
gcter_facet, 110
gcter_fallback_replication_configuration,
304
gcter_feature, 146
gcter_federation_token, 238, 871
gcter_feedback, 552
gcter_field_level_encryption, 118
gcter_field_level_encryption_config, 118
gcter_field_level_encryption_profile, 118
gcter_field_level_encryption_profile_config,
118
gcter_file, 180
gcter_file_upload_url, 580
gcter_filter, 436
gcter_finding, 10
gcter_finding_aggregator, 800
gcter_finding_history, 800
gcter_finding_statistics, 558
gcter_finding_v2, 10
gcter_findings, 193, 436, 557, 800
gcter_findings_filter, 557
gcter_findings_publication_configuration,
INDEX

get_findings_report_account_summary, 188
get_findings_report_status, 464
get_findings_statistics, 436
get_firewall_config, 754
get_firewall_domain_list, 754
get_firewall_rule_group, 754
get_firewall_rule_group_association, 754
get_firewall_rule_group_policy, 754
get_flow_association, 238
get_flow_logs_integration_template, 325
get_folder, 180, 930
get_folder_path, 930
get_form_type, 274
get_foundation_model, 90
get_function, 118, 515
get_function_code_signing_config, 515
get_function_concurrency, 515
get_function_configuration, 515
get_function_event_invoke_config, 515
get_function_url_config, 515
get_gateway, 82
get_gateway_response, 22
get_gateway_responses, 22
get_gcm_channel, 654
get_generated_policy, 10
get_generated_template, 114
get_geo_location, 738
get_geo_match_set, 913, 917
get_geofence, 545
get_global_settings, 925
get_glossary, 274
get_glossary_term, 274
get_grant, 531
get_gremlin_query_status, 590
get_group, 216, 446, 732, 883, 949
get_group_configuration, 732
get_group_id, 454
get_group_membership_id, 454
get_group_policy, 446
get_group_profile, 274
get_group_query, 732
get_groups, 949
get_groups_for_capacity_reservation, 325
get_guardrail, 90
get_health_check, 738
get_health_check_count, 738
get_health_check_last_failure_reason, 738
get_health_check_status, 738
get_health_event, 149
get_hit, 580
get_host, 205
get_host_reservation_purchase_preview, 325
get_hosted_zone, 738
get_hosted_zone_count, 738
get_hosted_zone_limit, 738
get_hostname_suggestion, 619
get_hypervisor, 82
get_hypervisor_property_mappings, 82
get_iam_portal_login_url, 274
get_id, 211
get_id_mapping_job, 379
get_id_mapping_workflow, 379
get_id_namespace, 379
get_identity_dkim_attributes, 820
get_identity_mail_from_domain_attributes, 820
get_identity_notification_attributes, 820
get_identity_policies, 820
get_identity_pool_configuration, 219
get_identity_pool_roles, 211
get_identity_provider, 946
get_identity_provider_by_identifier, 216
get_identity_resolution_job, 267
get_identity_source, 904
get_identity_verification_attributes, 820
get_image, 457
get_image_block_public_access_state, 325
get_image_pipeline, 457
get_image_policy, 457
get_image_recipe, 457
get_image_recipe_policy, 457
get_impersonation_role, 936
get_impersonation_role_effect, 936
get_import, 133, 519
get_import_job, 254, 654, 824
get_import_jobs, 654
get_in_app_messages, 654
get_in_app_template, 654
get_incident_record, 831
get_index, 729
get_infrastructure_configuration, 457
get_ingestion, 32
get_ingestion_destination, 32
get_inline_policy_for_permission_set, 860
get_insight, 949
get_insight_events, 949
get_insight_impact_graph, 949
get_insight_results, 800
get_insight_rule_report, 139
get_insight_selectors, 133
get_insight_summaries, 949
get_insight_impact_graph, 949
get_insight_results, 800
get_insight_rule_report, 139
get_insight_summaries, 949
get_insight_impact_graph, 949
get_insight_results, 800
get_insight_rule_report, 139
get_insight_summaries, 949
get_insight_impact_graph, 949
get_insight_results, 800
get_insight_rule_report, 139
get_insight_summaries, 949
get_insight_impact_graph, 949
get_insight_results, 800
get_insight_rule_report, 139
get_insight_summaries, 949
get_insight_impact_graph, 949
get_insight_results, 800
get_integrations, 29
get_integration, 22, 28, 267
get_integration_response, 22, 29
get_integration_responses, 29
get_integrations, 29
get_intent, 519
get_intent_versions, 519
get_intents, 519
get_internet_event, 149
get_invocation, 118
get_inventory, 844
get_inventory_schema, 844
get_investigation, 282
get_invitations_count, 436, 558, 800
get_ip_access_settings, 946
get_ip_set, 436, 913, 917, 922
get_ipam_address_history, 325
get_ipam_discovered_accounts, 325
get_ipam_discovered_public_addresses, 325
get_ipam_discovered_resource_cidrs, 325
get_ipam_pool_allocations, 325
get_ipam_pool_cidrs, 325
get_ipam_resource_cidrs, 325
get_item, 307
get_job, 98, 428
get_job_bookmark, 428
get_job_details, 198
get_job_output, 419
get_job_run, 376, 428
get_job_runs, 428
get_job_tagging, 762
get_jobs, 428
get_journey, 654
get_journey_date_range_kpi, 654
get_journey_execution_activity_metrics, 654
get_journey_execution_metrics, 654
get_journey_run_execution_activity_metrics, 654
get_journey_run_execution_metrics, 654
get_journey_runs, 654
get_job_tagging, 762
get_job_tagging, 762
get_key, 633
get_key_access_settings, 946
get_key_group, 118
get_key_group_config, 118
get_key_pair, 541
get_key_pairs, 541
get_key_policy, 507
get_key_rotation_status, 507
get_keyspace, 495
get_kms_encryption_key, 412
get_kx_changeset, 391
get_kx_cluster, 391
get_kx_connection_string, 391
get_kx_database, 391
get_kx_dataview, 391
get_kx_environment, 391
get_kx_scaling_group, 391
get_kx_user, 391
get_kx_volume, 391
INDEX

get_label_detection, 722
get_labels, 412
get_lambda_function_recommendations, 229
get_landing_zone, 258
get_landing_zone_operation, 258
get_launch, 146
get_launch_configuration, 304
get_launch_profile, 601
get_launch_profile_details, 601
get_launch_profile_initialization, 601
get_launch_profile_member, 601
get_launch_template_data, 325
get_layer_version, 515
get_layer_version_by_arn, 515
get_layer_version_policy, 515
get_layout, 247
get_legal_hold, 78
get_lending_analysis, 889
get_lending_analysis_summary, 889
get_lens, 925
get_lens_review, 925
get_lens_review_report, 925
get_lens_version_difference, 925
get_lexicon, 668
get lf_tag, 511
get_license, 531
get_license_configuration, 531
get_license_conversion_task, 531
get_license_manager_report_generator, 531
get_license_recommendations, 229
get_license_usage, 531
get_lifecycle_execution, 457
get_lifecycle_policies, 295
get_lifecycle_policy, 295, 334, 457
get_lifecycle_policy_preview, 334
get_lineage_group_policy, 772
get_link, 156
get_link_associations, 598
get_link_attributes, 110
get_links, 598
get_list_elements, 412
get_listener, 910
get_listing, 274
get_lists_metadata, 412
get_load_balancer, 541
get_load_balancer_metric_data, 541
get_load_balancer_tls_certificates, 541
get_load_balancer_tls_policies, 541
get_load_balancers, 541
get_loader_job_status, 590
get_log_anomaly_detector, 153
get_log_delivery_configuration, 216
get_log_events, 153
get_log_group_fields, 153
get_log_record, 153
get_logging_configuration, 475, 913, 917, 922
get_login_profile, 446
get_macie_session, 558
get_mail_domain, 936
get_mailbox_details, 936
get_maintenance_window, 844
get_maintenance_window_execution, 844
get_maintenance_window_execution_task, 844
get_maintenance_window_execution_task_invocation, 844
get_maintenance_window_task, 844
get_malware_scan_settings, 436
get_managed_endpoint_session_credentials, 374
get_managed_prefix_list_associations, 325
get_managed_prefix_list_entries, 325
get_managed_resource, 57
get_managed_rule_set, 922
get_managed_scaling_policy, 370
get_map_glyphs, 545
get_map_sprites, 545
get_map_style_descriptor, 545
get_map_tile, 545
get_mapping, 428
get_master_account, 436, 558, 800
get_match_id, 379
get_matches, 267
get_matching_job, 379
get_matching_workflow, 379
get_media_analysis_job, 722
get_medical_scribe_job, 897
get_medical_transcription_job, 897
get_medical_vocabulary, 897
get_member, 464, 538
get_member_detectors, 436
get_members, 282, 436, 800
get_merge_commit, 180
get_merge_conflicts, 180
get_merge_options, 180
get_message_insights, 824
get_metadata_generation_run, 274
get_method, 22
get_method_response, 22
get_metric_data, 139, 239
get_metric_data_v2, 239
get_metric_statistics, 139
get_metric_stream, 139
get_metric_widget_image, 139
get_metrics_summary, 193
get_mfa_device, 446
get_migration, 519
get.Migrations, 519
get_milestone, 926
get.ml_data_processing_job, 590
get_ml_endpoint, 590
get_ml_model, 555
get_ml_model_training_job, 590
get_ml_model_transform_job, 590
get_ml_task_run, 428
get_ml_task_runs, 428
get_ml_transform, 428
get_ml_transforms, 428
get_mobile_device_access_effect, 936
get_mobile_device_access_override, 936
get_mobile_sdk_release, 922
get_model, 22, 29
get_model_customization_job, 90
get_model_invocation_logging_configuration, 90
get_model_package_group_policy, 772
get_model_template, 22, 29
get_model_version, 412
get_models, 22, 29, 412
get_monitor, 149
get_monitoring_subscription, 118
get_multi_region_access_point, 762
get_multi_region_access_point_policy, 762
get_multi_region_access_point_policy_status, 762
get_multi_region_access_point_routes, 762
get_named_query, 60
get_namespace, 716, 814
get_network_insights_access_scope_analysis_findings, 325
get_network_insights_access_scope_content, 325
get_network_resource_counts, 598
get_network_resource_relationships, 598
get_network_resources, 598
get_network_routes, 598
get_network_settings, 946
get_network_telemetry, 598
get_notebook_metadata, 60
get_notification_channel, 402
get_notification_configuration, 188
get_object, 758
get_object_acl, 758
get_object_attributes, 110, 758
get_object_information, 110
get_object_legal_hold, 758
get_object_lock_configuration, 758
get_object_metadata, 84
get_object_retention, 758
get_object_tagging, 758
get_object_torrent, 758
get_on_premises_instance, 185
get_open_cypher_query_status, 590
get_open_id_connect_provider, 446
get_open_id_token, 211
get_open_id_token_for_developer_identity, 211
get_operation, 541, 814, 854
get_operation_detail, 741
get_operations, 541
get_operations_for_resource, 541
get_ops_item, 844
get_ops_metadata, 844
get_ops_summary, 844
get_organization_admin_account, 65
get_organization_config_rule_detailed_status, 233
get_organization_conformance_pack_detailed_status, 233
get_organization_custom_rule_policy, 233
get_organization_statistics, 436
get_organizations_access_report, 446
get_origin_access_control, 118
index

get_origin_access_control_config, 118
get_origin_request_policy, 118
get_origin_request_policy_config, 118
get_outcomes, 412
get_outpost_resolver, 754
get_package_version_asset, 165
get_package_version_history, 360, 612
get_package_version_readme, 165
get_parallel_data, 900
get_parameter, 844
get_parameter_history, 844
get_parameters, 845
get_parameters_by_path, 845
get_parameters_for_export, 633
get_parameters_for_import, 507, 633
get_participant, 480
get_partition, 428
get_partition_indexes, 428
get_partitions, 428
get_password_data, 325
get_patch_baseline, 845
get_patch_baseline_for_patch_group, 845
get_performance_analysis_report, 650
get_permission, 694
get_permission_group, 394
get_permission_policy, 913, 917, 922
get_permissions_boundary_for_permission_set, 860
get_person_tracking, 722
get_personalized_ranking, 647
get_pipeline, 198, 608
get_pipeline_blueprint, 608
get_pipeline_change_progress, 608
get_pipeline_definition, 271
get_pipeline_execution, 198
get_pipeline_state, 198
get_place, 545
get_plan, 428
get_platform_application_attributes, 837
get_playback_key_pair, 470
get_playbackrestriction_policy, 471
get_policies_stats, 615
get_policy, 18, 188, 379, 402, 446, 515, 904
get_policy_store, 904
get_policy_template, 904
get_policy_version, 446
get_portal, 946
get_portal_service_provider_metadata, 946
get_predictive_scaling_forecast, 72
get_prepared_statement, 60
get_price_list_file_url, 670
get_principal_tag_attribute_map, 211
get_products, 670
get_profile, 188, 451, 926
get_profile_object_type, 267
get_profileobjecttype_template, 268
get_profile_template, 926
get_programmatic_access_credentials, 394
get_project, 146, 173, 274
get_prompt_file, 239
get_protocol_list, 402
get_provider_service, 379
get_provisioned_concurrency_config, 515
get_provisioned_model_throughput, 90
get_provisioned_product_outputs, 811
get_public_access_block, 758, 762
get_public_key, 118, 507
get_public_key_certificate, 633
get_public_key_config, 118
get_pull_request, 180
get_pull_requestapproval_states, 180
get_pull_requestoverride_state, 180
get_push_template, 654
get_qualification_score, 580
get_qualification_type, 580
get_quantum_task, 98
get_query_execution, 60
get_querylogging_config, 738
get_queryresults, 60, 133, 149, 153
get_queryruntime_statistics, 60
get_querystate, 511
get_query_statistics, 511
get_query_status, 149
get_query_suggestions, 489
get_queue_attributes, 840
get_queue_url, 840
get_quick_response, 254
get_random_password, 795
get_raster_data_collection, 782
get_rate_based_rule, 913, 917
get_rate_based_ruleManagedKeys, 913, 917
get_rate_based_statementManagedKeys, 922
get_raw_message_content, 939
get_rdf_graph_summary, 591
get_read_set, 605
get_read_set_activation_job, 605
get_read_set_export_job, 605
get_read_set_import_job, 605
get_read_set_metadata, 605
get_readiness_check, 750
get_readiness_check_resource_status, 750
get_readiness_check_status, 750
get_realtime_log_config, 118
get_recommendation_preferences, 229
get_recommendation_summaries, 230
get_recommendations, 188, 254, 647
get_recommender_configuration, 654
get_recommender_configurations, 654
get_record, 780
get_recording_configuration, 471
get_records, 311, 497
get_recovery_group, 750
get_recovery_group_readiness_summary, 750
get_recovery_point, 716
get_recovery_point_restore_metadata, 78
get_reference, 605
get_reference_import_job, 605
get_reference_metadata, 605
get_reference_store, 605
get_regex_match_set, 913, 917
get_regex_pattern_set, 913, 922
get_region_opt_status, 13
get_regions, 541
get_registry, 428
get_registry_catalog_data, 337
get_registry_policy, 334
get_registry_scanning_configuration, 334
get_relational_database, 541
get_relational_database_blueprints, 541
get_relational_database_bundles, 541
get_relational_database_events, 541
get_relational_database_log_events, 541
get_relational_database_log_streams, 541
get_relational_database_master_user_password, 541
get_relational_database_metric_data, 541
get_relational_database_parameters, 541
get_relational_database_snapshot, 541
get_relational_database_snapshots, 541
get_relational_databases, 541
get_remaining_free_trial_days, 436
get_replication_configuration, 304
get_replication_set, 851
get_repository, 180, 678
get_repository_catalog_data, 337
get_repository_endpoint, 165
get_repository_link, 205
get_repository_permissions_policy, 165
get_repository_policy, 334, 337
get_repository_sync_status, 205, 678
get_repository_triggers, 180
get_request_validator, 22
get_request_validators, 22
get_requested_service_quota_change, 817
get_reservation_coverage, 264
get_reservation_purchase_recommendation, 264
get_reservation_utilization, 264
get_reserved_instances_exchange_quote, 325
get_reserved_node_exchange_configuration_options, 710
get_reserved_node_exchange_offers, 710
get_resolvers_config, 754
get_resolvers_dnssec_config, 754
get_resolvers_endpoint, 754
get_schema_version, 428
get_schema_versions_diff, 428
get_sdk, 22
get_sdk_type, 22
get_sdk_types, 22
get_search_suggestions, 772
get_secret_value, 795
get_security_config, 615
get_security_configuration, 428
get_security_configurations, 428
get_security_control_definition, 800
get_security_groups_for_vpc, 325
get_security_policy, 615
get_segment, 146, 654
get_segment_detection, 722
get_segment_export_jobs, 654
get_segment_import_jobs, 654
get_segment_version, 654
get_segments, 654
get_send_quota, 820
get_send_statistics, 820
get_sensitive_data_occurrences, 558
get_sensitive_data_occurrences_availability, 558
get_sensitivity_inspection_template, 558
get_sequence_store, 605
get_serial_console_access_status, 325
get_server_certificate, 446
get_service, 678, 814, 910
get_service_graph, 949
get_service_instance, 678
get_service_instance_sync_status, 678
get_service_last_accessed_details, 446
get_service_last_accessed_details_with_entity, 447
get_service_linked_role_deletion_status, 447
get_service_network, 910
get_service_network_service_association, 910
get_service_network_vpc_association, 910
get_service_principal_name, 639
get_service_quota, 817
get_service_quota_increase_request_from_template, 817
get_service_setting, 845
get_service_settings, 531, 534
get_service_sync_blocker_summary, 678
get_service_sync_config, 678
get_service_template, 678
get_service_template_version, 678
get_services_in_scope, 65
get_session, 60, 255, 428, 526, 528
get_session_embed_url, 689
get_session_status, 60
get_session_token, 871
get_settings, 65
get_setup_history, 541
get_shard_iterator, 311, 498
get_share, 605
get_signing_certificate, 216
get_similar_profiles, 268
get_sink, 156
get_sink_policy, 156
get_site_to_site_vpn_attachment, 598
get_sites, 598
get_size_constraint_set, 913, 918
get_slot_type, 519
get_slot_type_versions, 519
get_slot_types, 519
get_sms_attributes, 837
get_sms_channel, 654
get_sms_sandbox_account_status, 837
get_sms_template, 654
get_snapshot, 716
get_snapshot_block, 313
get_snapshot_block_public_access_state, 325
get_snapshot_limits, 292
get_snapshots, 489
get_solution_metrics, 642
g et_solution_package, 886
get_solution_package_content, 886
get_solution_package_descriptor, 886
g et_sol_function_instance, 886
get_sol_function_package, 886
get_sol_function_package_content, 886
get_sol_function_package_descriptor, 886
get_sol_network_instance, 886
get_sol_network_operation, 886
get_sol_network_package, 886
get_sol_network_package_content, 886
get_sol_network_package_descriptor, 886
get_source_repository, 173
get_source_repository_clone_urls, 173
get_space, 173
get_sparql_statistics, 591
get_sparql_stream, 591
get_speech_synthesis_task, 668
get_spot placement_scores, 325
get_sql_injection_match_set, 913, 918
get_ssh_public_key, 447
get_stack_policy, 114
get_stage, 22, 29, 480
get_stage_session, 480
get_stages, 22, 29
get_statement, 428
get_statement_result, 713
get_static_ip, 541
get_static_ips, 541
get_storage_configuration, 480
get_storage_lens_configuration, 762
get_storage_lens_configuration_tagging, 762
get_storage_lens_group, 762
get_stored_query, 233
get_stream, 471
get_stream_key, 471
get_stream_session, 471
get_streaming_distribution, 118
get_streaming_distribution_config, 118
get_streaming_image, 601
get_streaming_session, 602
get_streaming_session_backup, 602
get_streaming_session_stream, 602
get_studio, 602
get_studio_component, 602
get_studio_member, 602
get_studio_session_mapping, 370
get_subject, 451
get_subnet_cidr_reservations, 325
get_subscriber, 804
get_subscription, 173, 275
get_subscription_attributes, 837
get_subscription_grant, 275
get_subscription_request_details, 275
get_subscription_state, 831
get_subscription_target, 275
get_supported_resource_types, 78
get_suppressed_destination, 824
get_sync_blocker_summary, 205
get_sync_configuration, 205
get_table, 428, 495
get_table_auto_scaling_settings, 495
get_table_metadata, 60
get_table_objects, 511
get_table_optimizer, 428
get_table_restore_status, 716
get_table_version, 428
g...
<table>
<thead>
<tr>
<th>Method</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>get_trail_status</td>
<td>133</td>
</tr>
<tr>
<td>get_transcript</td>
<td>252</td>
</tr>
<tr>
<td>get_transcription_job</td>
<td>898</td>
</tr>
<tr>
<td>get_transit_gateway_attachment_propagations</td>
<td>325</td>
</tr>
<tr>
<td>get_transit_gateway_connect_peer_associations</td>
<td>598</td>
</tr>
<tr>
<td>get_transit_gateway_multicast_domain_associations</td>
<td>325</td>
</tr>
<tr>
<td>get_transit_gateway_peering</td>
<td>598</td>
</tr>
<tr>
<td>get_transit_gateway_policy_table_associations</td>
<td>325</td>
</tr>
<tr>
<td>get_transit_gateway_policy_table_entries</td>
<td>325</td>
</tr>
<tr>
<td>get_transit_gateway_prefix_list_references</td>
<td>325</td>
</tr>
<tr>
<td>get_transit_gateway_registrations</td>
<td>598</td>
</tr>
<tr>
<td>get_transit_gateway_route_table_associations</td>
<td>325</td>
</tr>
<tr>
<td>get_transit_gateway_route_table_attachment</td>
<td>598</td>
</tr>
<tr>
<td>get_transit_gateway_route_table_propagations</td>
<td>325</td>
</tr>
<tr>
<td>get_trigger</td>
<td>428</td>
</tr>
<tr>
<td>get_triggers</td>
<td>428</td>
</tr>
<tr>
<td>get_trust_anchor</td>
<td>451</td>
</tr>
<tr>
<td>get_trust_store</td>
<td>946</td>
</tr>
<tr>
<td>get_trust_store_ca_certificates_bundle</td>
<td>367</td>
</tr>
<tr>
<td>get_trust_store_certificate</td>
<td>946</td>
</tr>
<tr>
<td>get_trust_store_revocation_content</td>
<td>367</td>
</tr>
<tr>
<td>get_typed_link_facet_information</td>
<td>110</td>
</tr>
<tr>
<td>get_ui_customization</td>
<td>216</td>
</tr>
<tr>
<td>get_unfiltered_partition_metadata</td>
<td>428</td>
</tr>
<tr>
<td>get_unfiltered_partitions_metadata</td>
<td>428</td>
</tr>
<tr>
<td>get_unfiltered_table_metadata</td>
<td>428</td>
</tr>
<tr>
<td>get_upgrade_history</td>
<td>360</td>
</tr>
<tr>
<td>get_upgrade_status</td>
<td>360</td>
</tr>
<tr>
<td>get_usage</td>
<td>22</td>
</tr>
<tr>
<td>get_usage_forecast</td>
<td>264</td>
</tr>
<tr>
<td>get_usage_limit</td>
<td>716</td>
</tr>
<tr>
<td>get_usage_plan</td>
<td>22</td>
</tr>
<tr>
<td>get_usage_plan_key</td>
<td>22</td>
</tr>
<tr>
<td>get_usage_plan_keys</td>
<td>22</td>
</tr>
<tr>
<td>get_usage_plans</td>
<td>22</td>
</tr>
<tr>
<td>get_usage_statistics</td>
<td>437</td>
</tr>
<tr>
<td>get_usage_totals</td>
<td>558</td>
</tr>
<tr>
<td>get_user</td>
<td>216</td>
</tr>
<tr>
<td>get_user_access_logging_settings</td>
<td>447</td>
</tr>
<tr>
<td>get_user_attribute_verification_code</td>
<td>216</td>
</tr>
<tr>
<td>get_user_defined_function</td>
<td>428</td>
</tr>
<tr>
<td>get_user_defined_functions</td>
<td>428</td>
</tr>
<tr>
<td>get_user_details</td>
<td>173</td>
</tr>
<tr>
<td>get_user_endpoints</td>
<td>654</td>
</tr>
<tr>
<td>get_user_id</td>
<td>434</td>
</tr>
<tr>
<td>get_user_pool_mfa_config</td>
<td>216</td>
</tr>
<tr>
<td>get_user_profile</td>
<td>275</td>
</tr>
<tr>
<td>get_user_settings</td>
<td>947</td>
</tr>
<tr>
<td>get_utterances_view</td>
<td>519</td>
</tr>
<tr>
<td>get_variables</td>
<td>412</td>
</tr>
<tr>
<td>get_variant_import_job</td>
<td>605</td>
</tr>
<tr>
<td>get_variant_store</td>
<td>605</td>
</tr>
<tr>
<td>get_vault_access_policy</td>
<td>419</td>
</tr>
<tr>
<td>get_vault_lock</td>
<td>419</td>
</tr>
<tr>
<td>get_vault_notifications</td>
<td>419</td>
</tr>
<tr>
<td>get_vector_enrichment_job</td>
<td>782</td>
</tr>
<tr>
<td>get_verified_access_endpoint_policy</td>
<td>325</td>
</tr>
<tr>
<td>get_verified_access_group_policy</td>
<td>325</td>
</tr>
<tr>
<td>get_view</td>
<td>729</td>
</tr>
<tr>
<td>get_violation_details</td>
<td>402</td>
</tr>
<tr>
<td>get_virtual_machine</td>
<td>82</td>
</tr>
<tr>
<td>get_vocabulary</td>
<td>898</td>
</tr>
<tr>
<td>get_vocabulary_filter</td>
<td>898</td>
</tr>
<tr>
<td>get_vpc_channel</td>
<td>654</td>
</tr>
<tr>
<td>get_vpc_link</td>
<td>22</td>
</tr>
<tr>
<td>get_vpc_links</td>
<td>22</td>
</tr>
<tr>
<td>get_vpn_connection_device_sample_configuration</td>
<td>325</td>
</tr>
<tr>
<td>get_vpn_connection_device_device_types</td>
<td>325</td>
</tr>
<tr>
<td>get_vpn_tunnel_replacement_status</td>
<td>325</td>
</tr>
<tr>
<td>get_web_acl</td>
<td>914</td>
</tr>
<tr>
<td>get_web_acl_for_resource</td>
<td>918</td>
</tr>
<tr>
<td>get_work_group</td>
<td>60</td>
</tr>
<tr>
<td>get_work_unit_results</td>
<td>511</td>
</tr>
<tr>
<td>get_work_units</td>
<td>511</td>
</tr>
<tr>
<td>get_workflow</td>
<td>173</td>
</tr>
<tr>
<td>get_workflow_execution</td>
<td>457</td>
</tr>
<tr>
<td>get_workflow_execution_history</td>
<td>880</td>
</tr>
<tr>
<td>get_workflow_run</td>
<td>173</td>
</tr>
<tr>
<td>get_web_acls</td>
<td>918</td>
</tr>
<tr>
<td>get_web_acl_for_resource</td>
<td>922</td>
</tr>
</tbody>
</table>
INDEX

get_workflow_run_properties, 428
get_workflow_runs, 428
get_workflow_step_execution, 457
get_workflow_steps, 268
get_workgroup, 716
get_working_location, 394
get_workload, 926
get_xss_match_set, 914, 918
glacier, 416
global_sign_out, 216
globalaccelerator, 419
glue, 424
gluedatabrew, 430
grant_access, 619
grant_permissions, 511
group_resources, 732
guardduty, 433
head_bucket, 758
head_object, 758
health, 437
healthlake, 441

iam, 443
iamrolesanywhere, 449
identitystore, 452
imagebuilder, 455
import_api, 29
import_api_keys, 22
import_application_usage, 38
import_as_provisioned_product, 811
import_catalog_to_glue, 428
import_certificate, 15
import_certificate_authority_certificate, 18
import_client_branding, 943
import_client_vpn_client_certificate_revocation_list, 325
import_component, 457
import_crl, 451
import_dataset, 548
import_documentation_parts, 22
import_firewall_domains, 754
import_hub_content, 772
import_hypervisor_configuration, 82
import_image, 325
import_instance, 325
import_key, 633
import_key_material, 507
import_key_pair, 325, 541
import_lens, 926
import_model, 223
import_model_version, 548
import_notebook, 60
import_phone_number, 239
import_playback_key_pair, 471
import_resources_to_draft_app_version, 726
import_rest_api, 22
import_snapshot, 325
import_source_credentials, 168
import_stacks_to_stack_set, 114
import_table, 307
import_terminology, 900
import_vm_image, 457
import_volume, 326
import_workspace_image, 943
increase_node_groups_in_global_replication_group, 351
increase_replica_count, 351
increase_replication_factor, 278
increase_stream_retention_period, 498
index_documents, 128
index_faces, 722
infer_icd10cm, 226
infer_rx_norm, 226
infer_snomedct, 226
initialize_cluster, 125
initialize_service, 304
initiate_auth, 216
initiate_document_version_upload, 930
initiate_job, 419
initiate_layer_upload, 334, 337
initiate_multipart_upload, 419
initiate_vault_lock, 419
inspector, 458
inspector2, 462
instantiate_sol_network_instance, 886
invalidate_project_cache, 168
invite_account_to_organization, 627
invite_members, 437, 800
invoke, 515
invoke_async, 515
invoke_endpoint, 787
invoke_endpoint_async, 787
invoke_endpoint_with_response_stream, 787
invoke_model, 92
invoke_model_with_response_stream, 92
invoke_with_response_stream, 515
is_authorized, 904
is_authorized_with_token, 904
is_member_in_groups, 454
is_vpc_peered, 541
issue_certificate, 18
ivs, 465
ivschat, 471
ivsrealtime, 476
join_domain, 868
kafka, 481
kafkaconnect, 484
kendra, 487
kendraranking, 490
keyspaces, 492
kinesis, 495
kinesisanalytics, 498
kinesisanalyticsv2, 501
kms, 504
label_parameter_version, 845
lakeformation, 508
lambda, 512
leave_organization, 627
lexmodelbuildingservice, 516
lexmodelsv2, 519
lexruntimeservice, 524
lexruntimev2, 526
licensemanager, 529
licensemanagerlinuxsubscriptions, 532
licensemanagerusersubscriptions, 534
lightsail, 537
list_accelerators, 423
list_accepted_portfolio_shares, 811
list_access_control_configurations, 489
list_access_control_rules, 936
list_access_entries, 347
list_access_grants, 762
list_access_grants_instances, 762
list_access_grants_locations, 762
list_access_keys, 447
list_access_log_subscriptions, 910
list_access_points, 762
list_access_points_for_object_lambda, 762
list_access_policies, 347, 615
list_access_preview_findings, 10
list_access_previews, 10
list_access_tokens, 173
list_account_aliases, 447
list_account_assignment_creation_status, 860
list_account_assignment_deletion_status, 860
list_account_assignments, 860
list_account_assignments_for_principal, 860
list_account_associations, 95
list_account_integrations, 268
list_account_links, 943
list_account_permissions, 464
list_account_roles, 857
list_account_settings, 340
list_accounts, 627, 857
list_accounts_for_parent, 627
list_accounts_for_provisioned_permission_set, 860
list_action_executions, 198
list_action_types, 198
list_actions, 399, 772
list_activated_rules_in_rule_group, 914, 918
list_activities, 828
list_activity_types, 880
list_adapter_versions, 889
list_adapters, 889
list_addons, 347
list_admin_accounts_for_organization, 402
list_admins_managing_account, 402
list_agent_statuses, 239
list_aggregate_discovered_resources, 233
list_aggregated_utterances, 522
list_alarm_recommendations, 726
list_alerts, 552
list_algorithms, 772
list_aliases, 508, 515, 633, 772, 936
list_allow_lists, 558
list_allowed_node_type_modifications, 351
INDEX

list_allowed_node_type_updates, 575
list_allowed_repositories_for_group, 165
list_analyses, 689
list_analytics_data_associations, 239
list_analyzed_resources, 10
list_analyzers, 10
list_annotation_import_jobs, 605
list_annotation_store_versions, 605
list_annotation_stores, 605
list_anomalies, 153
list_anomalies_for_insight, 285
list_anomalous_log_groups, 285
list_anomaly_detectors, 552
list_anomaly_group_related_metrics, 552
list_anomaly_group_summaries, 552
list_anomaly_group_time_series, 552
list_answers, 926
list_api_destinations, 142, 383
list_api_keys, 922
list_app_assessment_compliance_drifts, 726
list_app_assessments, 726
list_app_authorizations, 32
list_app_bundles, 32
list_app_component_compliances, 726
list_app_component_recommendations, 726
list_app_image_configs, 772
list_app_input_sources, 726
list_app_monitors, 159
list_app_version_app_components, 726
list_app_version_resource_mappings, 726
list_app_version_resources, 726
list_app_versions, 726
list_application_access_scopes, 860
list_application_assignments, 860
list_application_assignments_for_principal, 860
list_application_authentication_methods, 860
list_application_dependencies, 807
list_application_dpu_sizes, 60
list_application_grants, 860
list_application_instance_dependencies, 630
list_application_instance_node_instances, 630
list_application_instances, 630
list_application_providers, 860
list_application_revisions, 180
list_application_snapshots, 503
list_application_versions, 503, 807
list_applications, 41, 47, 185, 376, 500, 503, 807, 854, 860
list_applied_schema_arns, 110
list_approval_rule_templates, 180
list_approved_origins, 239
list_apps, 726, 772
list_apps_lists, 402
list_archive_rules, 10
list_archives, 142, 383
list_artifacts, 772
list_assessment_control_insights_by_control_domain, 65
list_assessment_framework_share_requests, 65
list_assessment_frameworks, 65
list_assessment_reports, 65
list_assessment_run_agents, 461
list_assessment_runs, 461
list_assessment_targets, 461
list_assessment_templates, 461
list_assessments, 65
list_asset_bundle_export_jobs, 690
list_asset_bundle_import_jobs, 690
list_asset_revisions, 275
list_assignments_for_hit, 580
list_assistant_associations, 255
list_assistants, 255
list_associated_access_policies, 347
list_associated_approval_rule_templates_for_repository, 180
list_associated_attribute_groups, 47
list_associated_fleets, 54
list_associated_groups, 883
list_associated_packages, 165
list_associated_resources, 47
list_associated_route_53_healthChecks, 747
list_associated_stacks, 54
list_association_versions, 845
list_associations, 772, 845
list_associations_for_license_configuration,
list_changed_blocks, 313
list_changesets, 394
list_channels, 133, 471
list_check_details, 926
list_check_summaries, 926
list_children, 627
list_chunks, 84
list_cidr_blocks, 738
list_cidr_collections, 738
list_cidr_locations, 738
list_cis_scan_configurations, 464
list_cis_scan_results_aggregated_by_checks, 464
list_cis_scan_results_aggregated_by_target_resource, 464
list_cis_scans, 464
list_classification_jobs, 558
list_classification_scopes, 558
list_client_vpc_connections, 483
list_closed_workflow_executions, 880
list_cloud_front_origin_access_ids, 119
list_cluster_nodes, 772
list_cluster_operations, 483
list_cluster_operations_v2, 483
list_cluster_snapshots, 301
list_clusters, 301, 340, 347, 370, 483, 747, 772
list_clusters_v2, 483
list_code_repositories, 773
list_code_reviews, 191
list_code_signing_configs, 515
list_collections, 615, 722
list_column_statistics_task_runs, 428
list_command_invocations, 845
list_commands, 845
list_compilation_jobs, 773
list_compliance_items, 845
list_compliance_status, 402
list_compliance_summaries, 845
list_component_build_versions, 457
list_component_outputs, 679
list_component_provisioned_resources, 679
list_components, 41, 457, 679, 854
list_compositions, 480
list_configuration_history, 41
list_configuration_policies, 800
list_configuration_policy_associations, 800
list_configuration_revisions, 483, 578
list_configuration_sets, 658, 661, 820, 824
list_configurations, 483, 578
list_conflicting_aliases, 119
list_conformance_pack_compliance_scores, 233
list_connect_peers, 598
list_connections, 50, 142, 205, 383
list_connectors, 486, 639
list_constraints_for_portfolio, 811
list_contact_channels, 848
list_contact_evaluations, 239
list_contact_flow_modules, 239
list_contact_flows, 239
list_contact_lists, 824
list_contact_references, 239
list_contacts, 824, 848
list_container_instances, 340
list_container_recipes, 457
list_contents, 255
list_contexts, 773
list_continuous_deployment_policies, 119
list_contributor_insights, 307
list_control_domain_insights, 65
list_control_domain_insights_by_assessment, 65
list_control_insights_by_control_domain, 65
list_control_panels, 747
list_controls, 65
list_copy_job_summaries, 78
list_copy_jobs, 78
list_core_network_policy_versions, 598
list_core_networks, 598
list_cost_allocation_tag_backfill_history, 264
list_cost_allocation_tags, 264
list_cost_category_definitions, 264
list_coverage, 437, 464
list_coverage_statistics, 464
list_crawlers, 428
list_crawls, 428
list_create_account_status, 627
list_crls, 451
list_cross_account_attachments, 423
list_cross_account_authorizations, 750
list_cross_account_resource_accounts, 423
list_cross_account_resources, 423
list_curated_environment_images, 168
list_custom_data_identifiers, 558
list_custom_domain_associations, 716
list_custom_entity_types, 428
list_custom_line_item_versions, 95
list_custom_line_items, 95
list_custom_models, 90
list_custom_plugins, 486
list_custom_routing_accelerators, 423
list_custom_routing_endpoint_groups, 423
list_custom_routing_listeners, 423
list_custom_routing_port_mappings, 423
list_custom_routing_port_mappings_by_destination, 423
list_custom_verification_email_templates, 820, 824
list_custom_vocabulary_items, 523
list_customer_managed_policy_references_in_permission_set, 679
list_dashboard_versions, 690
list_dashboards, 139, 690
list_data_catalogs, 60
list_data_cells_filter, 511
list_data_ingestion_jobs, 548
list_data_lake_exceptions, 804
list_data_lakes, 804
list_data_quality_job_definitions, 773
list_data_quality_results, 429
list_data_quality_rule_recommendation_runs, 429
list_data_quality_ruleset_evaluation_runs, 429
list_data_quality_rulesets, 429
list_data_sets, 690
list_data_source_run_activities, 275
list_data_source_runs, 275
list_data_source_sync_jobs, 489
list_data_sources, 275, 489, 612, 690
list_data_views, 394
list_databases, 60, 713, 854, 894
list_dataset_entries, 722
list_dataset_export_jobs, 642
list_dataset_groups, 408, 642
list_dataset_import_jobs, 408, 642
list_dataset_labels, 722
list_datasets, 220, 223, 394, 408, 433, 548, 642
list_datasource_packages, 282
list_dead_letter_source_queues, 840
list_dedicated_ip_pools, 658, 824
list_default_vocabularies, 239
list_delegated_admin_accounts, 464
list_delegated_administrators, 627
list_delegated_services_for_account, 627
list_deliverability_test_reports, 658, 824
list_delivery_streams, 397
list_deployment_configs, 185
list_deployment_groups, 185
list_deployment_instances, 185
list_deployment_targets, 185
list_deployments, 185, 679
list_detectors, 437
list_dev_endpoints, 429
list_dev_environment_sessions, 173
list_dev_environments, 173
list_device_fleets, 110
list_device_fleets, 773
list_devices, 216, 630, 773, 933
list_devices_jobs, 630
list_directories, 110
list_directory_buckets, 758
list_directory_registrations, 639
list_discovered_resources, 233, 402
list_discoverers, 792
list_distributed_grants, 531
list_distribution_configurations, 457
list_distributions, 119
list_distributions_by_cache_policy_id, 119
list_distributions_by_key_group, 119
list_distributions_by_origin_request_policy_id, 119
list_distributions_by_realtime_log_config, 119
list_distributions_by_response_headers_policy_id, 119
list_distributions_by_web_acl_id, 119
list_document_classification_jobs, 223
list_document_classifiers_summaries, 223
list_document_classifiers, 223
list_document_metadata_history, 845
list_document_versions, 845
list_documents, 845
list_domain_deliverability_campaigns, 658, 824
list_domain_maintenances, 612
list_domain_names, 128, 360, 612
list_domains, 165, 247, 268, 275, 741, 773, 834, 880, 907, 933
list_domains_for_package, 360, 612
list_dominant_language_detection_jobs, 223
list_earth_observation_jobs, 782
list_edge_deployment_plans, 773
list_eks_anywhere_subscriptions, 347
list_elasticsearch_instance_types, 360
list_elasticsearch_versions, 360
list_email_identities, 658, 824
list_email_templates, 824
list_enabled_baselines, 259
list_enabled_controls, 259
list_enabled_products_for_import, 800
list_encoder_configurations, 480
list_endpoint_access, 716
list_endpoint_configs, 773
list_endpoint_groups, 423
list_endpoints, 223, 383, 765, 773
list_endpoints_by_platform_application, 837
list_engagements, 849
list_engine_versions, 60
list_entities, 565
list_entities_detection_jobs, 223
list_entities_detection_v2_jobs, 226
list_entities_for_policy, 447
list_entitled_applications, 54
list_entity_personas, 489
list_entity_recognizer_summaries, 223
list_entity_recognizers, 223
list_environment_account_connections, 679
list_environment_blueprint_configurations, 275
list_environment_blueprints, 275
list_environment_outputs, 679
list_environment_profile, 275
list_environment_provisioned_resources, 679
list_environment_template_versions, 679
list_environment_template, 679
list_environment_templates, 679
list_environments, 102, 104, 275, 391, 584, 679
list_eula_acceptances, 602
list_eulas, 602
list_evaluation_form_versions, 239
list_evaluation_forms, 239
list_evaluation_jobs, 90
list_event_buses, 142, 383
list_event_data_stores, 133
list_event_logs, 173
list_event_predictions, 412
list_event_source_mappings, 515
list_event_sources, 142, 383
list_event_streams, 268
list_event_subscriptions, 461
list_event_trackers, 642
list_event_types, 208
list_events, 285
list_events_detection_jobs, 223
list_exclusions, 461
list_executions, 828
list_executors, 60
list_experience_entities, 489
list_experiences, 489
list_experiment_resolved_targets, 399
list_experiment_target_account_configurations, 399
list_experiment_target_account_configurations, 399
list_experiment_template, 399
list_experiment_templates, 146, 399, 773
list_explainabilities, 408
list_explainability_exports, 408
list_export_jobs, 824
list_exports, 114, 307, 523
list_extensible_source_servers, 304
list_faces, 722
list_facet_attributes, 110
list_facet_names, 110
list_failures_for_license_operation, 531
list_faqs, 489
list_fargate_profiles, 347
list_feature_groups, 773
list_featured_results_sets, 489
list_features, 146
list_fhir_datastores, 443
list_fhir_export_jobs, 443
list_fhir_import_jobs, 443
list_field_level_encryption_configs, 119
list_field_level_encryption_profiles, 119
list_field_options, 247
list_fields, 247
list_file_commit_history, 180
list_file_shares, 868
list_file_system_associations, 868
list_filters, 437, 464, 642
list_finding_aggregations, 464
list_finding_aggregators, 800
list_findings, 10, 437, 461, 464, 558
list_findings_filters, 558
list_findings_metrics, 193
list_findings_reports, 188
list_findings_v2, 10
list_firewall_configs, 754
list_firewall_domain_lists, 754
list_firewall_domains, 754
list_firewall_policies, 594
list_firewall_rule_group_associations, 754
list_firewall_rule_groups, 754
list_firewall_rules, 754
list_firewalls, 594
list_fleets, 168, 933
list_flow_associations, 239
list_flow_definitions, 773
list_flywheel_iteration_history, 223
list_flywheels, 223
list_folder_members, 690
list_folders, 690
list_forecast_export_jobs, 408
list_forecasts, 408
list_foundation_models, 90
list_frameworks, 78
list_fraudster_registration_jobs, 907
list_fraudsters, 907
list_function_event_invoke_configs, 515
list_function_url_configs, 515
list_functions, 119, 515
list_functions_by_code_signing_config, 515
list_gateway_routes, 44
list_gateways, 82, 868
list_generated_templates, 114
list_geo_locations, 738
list_geo_match_sets, 914, 918
list_geofence_collections, 545
list_geofences, 545
list_github_account_token_names, 185
list_global_tables, 307
list_grants, 508
list_graphs, 282
list_gremlin_queries, 591
list_group_members, 936
list_group_memberships, 454, 690
list_group_memberships_for_member, 454
list_group_policies, 447
list_group_resources, 732, 883
list_groups, 216, 447, 454, 690, 732, 884, 936
list_groups_for_entity, 936
list_groups_for_user, 447
list_groups_older_than_ordering_id, 489
list_guardrails, 90
list_handshakes_for_account, 627
list_handshakes_for_organization, 627
list_hapgs, 122
list_health_checks, 738
list_health_events, 149
list_hi_ts, 581
list_hi_ts_for_qualification_type, 581
list_hosted_zones, 738
list_hosted_zones_by_name, 738
list_hosted_zones_by_vpc, 738
list_hosts, 205
list_hours_of_operations, 239
list_hsms, 122
list_hub_content_versions, 773
list_hub_contents, 773
list_hubs, 773
list_human_loops, 68
list_human_task_uis, 773
list_hyperparameter_tuning_jobs, 773
list_hypervisors, 82
list_iam_policy_assignments, 690
list_iam_policy_assignments_for_user, 690
list_icd10cm_inference_jobs, 226
list_id_mapping_jobs, 379
list_id_namespaces, 379
list_identities, 211, 820
list_identity_policies, 820
list_identity_pool_usage, 220
list_identity_pools, 211
list_identity_propagation_configs, 690
list_identity_provider_configs, 347
list_identity_providers, 216, 536, 947
list_identity_resolution_jobs, 268
list_identity_sources, 904
list_image_build_versions, 457
list_image_packages, 457
list_image_pipeline_images, 457
list_image_pipelines, 457
list_image_recipes, 457
list_image_scan_finding_aggregations, 458
list_image_scan_findings, 458
list_image_versions, 773
list_images, 334, 457, 773
list_images_in_recycle_bin, 326
list_impersonation_roles, 936
list_import_failures, 133
list_import_jobs, 255, 824
list_imports, 114, 133, 307, 523
list_incident_findings, 852
list_incident_records, 852
list_incoming_typed_links, 110
list_index, 110
list_indexes, 729
list_indexes_for_members, 729
list_indicators, 282
list_indices, 489
list_inference_components, 773
list_inference_events, 548
list_inference_executions, 548
list_inference_experiments, 773
list_inference_recommendations_job_steps, 773
list_inference_recommendations_jobs, 773
list_inference_schedulers, 548
list_infrastructure_configurations, 458
list_ingestion_destinations, 32
list_ingerstions, 32, 690
list_insights, 285, 347
list_insights_metric_data, 133
list_instance_attributes, 239
list_instance_fleets, 370
list_instance_groups, 370
list_instance_profile_tags, 447
list_instance_profiles, 447
list_instance_profiles_for_role, 447
list_instance_storage_configs, 239
list_instance_type_details, 612
list_instances, 239, 370, 536, 814, 860
list_integration_associations, 239
list_integrations, 268
list_intent_metrics, 523
list_intent_paths, 523
list_intent_stage_metrics, 523
list_intents, 523
list_internet_events, 149
list_invalidations, 119
list_inventory_entries, 845
list_investigations, 282
list_invitations, 282, 437, 558, 800
list_ip_access_settings, 947
list_ip_routes, 292
list_ip_sets, 437, 914, 918, 922
list_job_runs, 374, 376, 433
list_job_templates, 374
list_jobs, 87, 419, 429, 433, 762
list_journal_kinesis_streams_for_ledger, 682
list_journal_s3_exports, 682
list_journal_s3_exports_for_ledger, 682
list_journeys, 654
list_kafka_versions, 483
list_key_groups, 119
list_key_phrases_detection_jobs, 223
list_key_policies, 508
list_key_rotations, 508
list_key_value_stores, 119
list_keys, 508, 545, 633
list_keyspaces, 495
list_keywords_for_data_source, 65
list_knowledge_bases, 255
list_kx_changesets, 391
list_kx_cluster_nodes, 391
list_kx_clusters, 391
list_kx_databases, 391
list_kx_dataviews, 391
list_kx_environments, 391
list_kx_scaling_groups, 391
list_kx_users, 391
list_kx_volumes, 391
list_label_groups, 549
list_labeling_jobs, 773
list_labeling_jobs_for_workteam, 773
list_labels, 549
list_lake_formation_opt_ins, 511
list_lambda_functions, 239
list_landing_zones, 259
list_language_models, 898
list_languages, 900
list_launch_actions, 304
list_launch_paths, 811
list_launch_profile_members, 602
list_launch_profiles, 602
list_launches, 146
list_layer_versions, 515
list_layers, 515
list_layouts, 247
list_ledgers, 682
list_legal_holds, 78
list_lens_review_improvements, 926
list_lens_reviews, 926
list_lens_shares, 926
list_lenses, 926
list_lex_bots, 239
list_lexicons, 668
list_lf_tags, 511
list_license_configurations, 531
list_license_conversion_tasks, 531
list_license_manager_report_generators, 531
list_license_specifications_for_resource, 531
list_license_versions, 531
list_licenses, 531
list_lifecycle_execution_resources, 458
list_lifecycle_executions, 458
list_lifecycle_policies, 458, 615
list_lineage_groups, 773
list_links, 156
list_linux_subscription_instances, 534
list_linux_subscriptions, 534
list_listeners, 423, 910
list_loader_jobs, 591
list_local_disks, 868
list_log_anomaly_detectors, 153
list_log_pattern_sets, 41
list_log_patterns, 41
list_log_sources, 804
list_log_subscriptions, 292
list_logging_configurations, 476, 914, 918, 922
list_luna_clients, 122
list_mail_domains, 936
list_mailbox_export_jobs, 936
list_mailbox_permissions, 936
list_managed_data_identifiers, 558
list_managed_endpoints, 374
list_managed_insight_rules, 139
list_managed_policies_in_permission_set, 860
list_managed_resources, 57
list.managed_rule_sets, 922
list.managed_schema_arns, 110
list_map_runs, 828
list_maps, 545
list_matching_jobs, 379
list_matching_workflows, 379
list_media_analysis_jobs, 722
list_medical_scribe_jobs, 898
list_medical_transcription_jobs, 898
list_medical_vocabulary, 898
list_member_accounts, 402
list_members, 282, 437, 464, 558, 800
list_meshes, 44
list_message_move_tasks, 840
list_metadata_generation_runs, 275
list_metric_attribution_metrics, 642
list_metric_attribution, 642
list_metric_sets, 552
list_metric_streams, 139
list_metrics, 139
list_mfa_device_tags, 447
list_mfa_devices, 447
list_milestones, 926
list_ml_data_processing_jobs, 591
list_ml_endpoints, 591
list_ml_model_training_jobs, 591
list_ml_model_transform_jobs, 591
list_ml_transforms, 429
list_mobile_device_access_overrides, 936
list_mobile_device_access_rules, 937
list_mobile_sdk_releases, 922
list_model_bias_job_definitions, 773
list_model_card_export_jobs, 773
list_model_card_versions, 773
list_model_cards, 773
list_model_customization_jobs, 90
list_model_explainability_job_definitions, 773
list_model_metadata, 773
list_model_package_groups, 773
list_model_packages, 773
list_model_quality_job_definitions, 773
list_model_versions, 549
list_models, 549, 773
list_monitor_evaluations, 408
list_monitored_resources, 285
list_monitoring_alert_history, 773
list_monitoring_alerts, 773
list_monitoring_executions, 773
list_monitoring_schedules, 773
list_monitors, 149, 408
list_multi_region_access_points, 762
list_multipart_read_set_uploads, 605
list_multipart_uploads, 419, 758
list_named_queries, 60
list_namespaces, 690, 716, 814
list_network_settings, 947
list_node_from_template_jobs, 630
list_nodegroups, 347
list_nodes, 483, 630
list_notebook_executions, 370
list_notebook_instance_lifecycle_configs, 773
list_notebook_instances, 773
list_notebook_metadata, 60
list_notebook_sessions, 60
list_notification_channels, 285
list_notification_rules, 208
list_notifications, 65, 275, 926
list_object_attributes, 110
list_object_children, 110
list_object_parent_paths, 110
list_object_parents, 110
list_object_policies, 110
list_object_versions, 758
list_objects, 84, 758
list_objects_v2, 758
list_observability_configurations, 50
list_on_premises_instances, 185
list_open_cypher_queries, 591
list_open_id_connect_provider_tags, 447
list_open_id_connect_providers, 447
list_open_workflow_executions, 880
list_operations, 50, 741, 814, 854
list_ops_item_events, 845
list_ops_item_related_items, 845
list_ops_metadata, 845
list_organization_admin_accounts, 282, 437, 558, 801
list_organization_insights, 285
list_organization_portfolio_access, 811
list_organization_service_access_status, 598
list_organizational_units_for_parent, 627
list_organizations, 937
list_origin_access_controls, 119
list_origin_request_policies, 119
list_origination_numbers, 837
list_outgoing.TypedLinks, 110
list_outpost_resolvers, 754
list_outposts_with_s3, 765
list_package_groups, 165
list_package_import_jobs, 630
list_package_version_assets, 165
list_package_version_dependencies, 165
list_package_versions, 165
list_packages, 165, 630
list_packages_for_domain, 360, 612
list_page_receipts, 849
list_page_resolutions, 849
list_pages_by_contact, 849
list_pages_by_engagement, 849
list_parallel_data, 900
list_parents, 627
list_participant_events, 480
list_participants, 480
list_partner_event_source_accounts, 142, 383
list_partner_event_sources, 142, 383
list_parts, 419, 578
list_peerings, 598
list_pending_invitation_resources, 694
list_performance_analysis_reports, 650
list_permission_associations, 694
list_permission_groups, 394
list_permission_groups_by_user, 394
list_permission_set_provisioning_status, 860
list_permission_sets, 860
list_permission_sets_provisioned_to_account, 861
list_permission_versions, 694
list_permissions, 18, 511, 561, 694
list_phi_detection_jobs, 226
list_phone_numbers, 239
list_phone_numbers_opted_out, 837
list_phone_numbers_v2, 239
list_pii_entities_detection_jobs, 223
list_pipeline_blueprints, 608
list_pipeline_execution_steps, 773
list_pipeline_executions, 198, 773
list_pipeline_parameters_for_execution, 773
list_pipelines, 198, 271, 608, 773
list_pipes, 386
list_place_indexes, 545
list_platform_applications, 837
list_platform_branches, 354
list_platform_versions, 354
list_playback_key_pairs, 471
list_playbackrestriction_policies, 471
list_pod_identity_associations, 347
list_policies, 402, 447, 627, 904
list_policies_for_target, 627
list_policies_granting_service_access, 447
list_policy_attachments, 110
list_policy_generations, 10
list_policy_stores, 904
list_policy_tags, 447
list_policy_templates, 904
list_policy_versions, 447
list_pool_origination_identities, 665
list_portals, 947
list_portfolio_access, 811
list_portfolios, 811
list_portfolios_for_product, 811
list_predefined_attributes, 239
list_predictor_backtest_export_jobs, 408
list_predictors, 408
list_prepared_statements, 60
list_preview_rotation_shifts, 849
list_price_lists, 670
list_prices, 741
list_pricing_plans, 95
list_pricing_plans_associated_with_pricing_rule, 95
list_pricing_rules, 95
list_pricing_rules_associated_to_pricing_plan, 95
list_principals, 694
list_principals_for_portfolio, 811
list_problems, 41
list_processing_jobs, 774
list_product_subscriptions, 536
list_profile_notifications, 926
list_profile_object_type_templates, 268
list_profile_object_types, 268
list_profile_objects, 268
list_profile_shares, 926
list_profile_times, 188
list_profiles, 451, 926
list_profiling_groups, 188
list_project_memberships, 275
list_project_policies, 722
list_projects, 146, 168, 173, 201, 275, 433, 774
list_prompts, 239
list_protected_resources, 78
list_protected_resources_by_backup_vault, 78
list_protection_groups, 831
list_protections, 831
list_protocols_lists, 402
list_provider_services, 380
list_provisioned_capacity, 419
list_provisioned_concurrency_configs, 515
list_provisioned_model_throughputs, 90
INDEX

list_provisioned_product_plans, 811
list_provisioning_artifacts, 811
list_provisioning_artifacts_for_service_action

list_public_keys, 119, 133
list_published_schema_arns, 110
list_publishing_destinations, 437
list_pull_requests, 180
list_qualification_requests, 581
list_qualification_types, 581
list_queries, 133
list_query_executions, 60
list_query_logging_configs, 738
list_query_suggestions_block_lists, 489
list_queue_quick_connects, 239
list_queue_tags, 840
list_queues, 239, 840
list_quick_connects, 239
list_quick_responses, 255
list_raster_data_collections, 782
list_rate_based_rules, 914, 918
list_read_set_activation_jobs, 605
list_read_set_export_jobs, 605
list_read_set_import jobs, 605
list_read_set_upload_parts, 605
list_read_sets, 605
list_readiness_checks, 750
list_readiness_checks, 750
list_read_time_contact_analysis_segments, 249
list_realtime_contact_analysis_segments, 249
list_realtime_contact_analysis_segments_v2, 249
list_realtime_log_configs, 119
list_receipt_filters, 820
list_receipt_rule_sets, 820
list_received_grants, 531
list_received_grants_for_organization, 531
list_received_licenses, 531
list_received_licenses_for_organization, 531
list_recipe_versions, 433
list_recipes, 433, 642
list_recommendation_feedback, 191
list_recommendation_templates, 726
list_recommendations, 191, 285, 710, 825
list_recommended_intents, 523
list_recommenders, 642
list_record_history, 811
list_recording_configurations, 471
list_records, 220
list_recovery_groups, 750
list_recovery_points, 716
list_recovery_points_by_backup_vault, 78
list_recovery_points_by_legal_hold, 78
list_recovery_points_by_resource, 78
list_reference_import_jobs, 605
list_reference_stores, 605
list_repositories, 165, 180, 679
list_repositories_for_approval_rule_template, 180
list_repositories_in_domain, 165
list_repository_associations, 191
list_repository_links, 205
list_repository_sync_definitions, 205, 679
list_requested_service_quota_change_history, 817
list_requested_service_quota_change_history_by_quota, 817
list_rescore_execution_plans, 492
list_resiliency_policies, 726
list_resilience_policies, 726
listresolver_configs, 754
list_resolver_dnssec_configs, 754
list_scrapers, 673
list_secret_version_ids, 795
list_secrets, 795
list_security_configs, 615
list_security_configurations, 370, 374
list_security_control_definitions, 801
list_security_keys, 239
list_security_policies, 615
list_security_profile_applications, 239
list_security_profile_permissions, 239
list_security_profiles, 239
list_segment_references, 146
list_segments, 146
list_sensitivity_inspection_templates, 558
list_sensor_statistics, 549
list_sentiment_detection_jobs, 223
list_sequence_stores, 605
list_server_certificate_tags, 447
list_server_certificates, 447
list_service_actions, 811
list_service_actions_for_provisioning_artifact, 811
list_service_instance_outputs, 679
list_service_instance_provisioned_resources, 679
list_service_instances, 679
list_service_network_service_associations, 910
list_service_network_vpc_associations, 910
list_service_networks, 910
list_service_pipeline_outputs, 679
list_service_pipeline_provisioned_resources, 679
list_service_principal_names, 639
list_service_quota_increase_requests_in_template, 817
list_service_quotas, 817
list_service_specific_credentials, 447
list_service_template_versions, 679
list_service_templates, 679
list_services, 50, 340, 679, 814, 817, 910
list_services_by_namespace, 340
list_services_for_auto_scaling_configuration, 50
list_session_analytics_data, 523
list_session_metrics, 523
list_sessions, 61, 429
list_shards, 498
list_share_invitations, 926
list_shared_endpoints, 765
list_shared_projects, 168
list_shared_report_groups, 168
list_shares, 606
list_signing_certificates, 447
list_sinks, 156
list_size_constraint_sets, 914, 918
list_slack_channel_configurations, 878
list_slack_workspace_configurations, 878
list_slot_types, 523
list_slots, 523
list_sms_sandbox_phone_numbers, 837
list_snapshot_blocks, 313
list_snapshot_copy_configurations, 716
list_snapshots, 716
list_snapshots_in_recycle_bin, 326
list_snomedct_inference_jobs, 226
list_sol_function_instances, 886
list_sol_function_packages, 886
list_sol_network_instances, 886
list_sol_network_operations, 886
list_sol_network_packages, 886
list_solution_versions, 643
list_solutions, 642
list_sop_recommendations, 726
list_source_credentials, 168
list_source_repositories, 173
list_source_repository_branches, 173
list_spaces, 173, 774
list_speaker_enrollment_jobs, 907
list_speakers, 907
list_speech_synthesis_tasks, 668
list_sql_injection_match_sets, 914, 918
list_ssh_public_keys, 447
list_stack_instance_resource_drifts, 114
list_stack_instances, 114
list_stack_instances_for_provisioned_product, 811
list_stack_resources, 114
list_stack_set_auto_deployment_targets, 114
list_stack_set_operation_results, 114
list_stack_set_operations, 114
list_stack_sets, 114
list_stacks, 114
list_stage_devices, 774
list_stage_sessions, 480
list_stages, 480
list_staging_accounts, 304
list_standards_control_associations, 801
list_state_machine_aliases, 828
list_state_machine_versions, 828
list_state_machines, 828
list_steps, 370
list_storage_configurations, 480
list_storage_lens_configurations, 762
list_storage_lens_groups, 762
list_stored_queries, 233
list_stream_consumers, 498
list_stream_keys, 471
list_stream_processors, 722
list_stream_sessions, 471
list_streaming_distributions, 119
list_streaming_images, 602
list_streaming_session_backups, 602
list_streaming_sessions, 602
list_streams, 311, 471, 498
list_studio_components, 602
list_studio_lifecycle_configs, 774
list_studio_members, 602
list_studio_session_mappings, 370
list_studios, 370, 602
list_sub_package_groups, 165
list_subjects, 451
list_subscribed_rule_groups, 914, 918
list_subscribed_workteams, 774
list_subscribers, 804
list_subscription_grants, 275
list_subscription_requests, 275
list_subscription_targets, 275
list_subscriptions, 275, 837
list_subscriptions_by_topic, 837
list_suggested_resiliency_policies, 726
list_supported_instance_types, 370
list_supported_resource_types, 729
list_suppressed_destinations, 825
list_sync_configurations, 205
list_table_metadata, 61
list_table_optimizer_runs, 429
list_table_restore_status, 716
list_table_storage_optimizers, 511
list_tables, 307, 495, 713, 894
list_tag_options, 811
list_tags, 18, 79, 125, 133, 278, 360, 515, 575, 578, 612, 619, 774
list_tags_for_certificate, 15
list_tags_for_delivery_stream, 397
list_tags_for_domain, 741
list_tags_for_project, 201
list_tags_for_resources, 738, 750
list_tags_for_stream, 498
list_tags_for_vault, 419
list_tags_log_group, 153
list_tags_of_resource, 307
list_tape_pools, 868
list_tapes, 868
list_target_account_configurations, 399
list_target_groups, 910
list_target_resource_types, 399
list_targeted_sentiment_detection_jobs,
INDEX

223
list_targets, 208, 910
list_targets_by_rule, 143, 383
list_targets_for_policy, 627
list_task_definition_families, 340
list_task_definitions, 340
list_tasks, 340
list_team_members, 201
list_template_aliases, 690
list_template_group_access_control_entries, 639
list_template_shares, 926
list_template_versions, 654, 690
list_templates, 247, 639, 654, 690, 820
list_terminologies, 900
list_test_execution_result_items, 523
list_test_executions, 523
list_test_recommendations, 726
list_test_set_records, 523
list_test_sets, 523
list_text_translation_jobs, 900
list_theme_aliases, 690
list_themes, 690
list_thesauri, 489
list_third_party_firewall_firewall_policies, 402
list_threat_intel_sets, 437
list_time_series_data_points, 275
list_timeline_events, 852
list_tls_inspection_configurations, 594
list_tokens, 531
list_topic_refresh_schedules, 690
list_topics, 690, 837
list_topics_detection_jobs, 223
list_tracker_consumers, 545
list_trackers, 545
list_traffic_distribution_group_users, 239
list_traffic_distribution_groups, 239
list_traffic_policies, 738
list_traffic_policy_instances, 738
list_traffic_policy_instances_by_hosted_zone, 738
list_traffic_policy_instances_by_policy, 738
list_traffic_policy_versions, 738
list_trails, 133
list_training_jobs, 774
list_training_jobs_for_hyper_parameter_tuning_job, 774
list_transactions, 511
list_transcription_jobs, 898
list_transform_jobs, 774
list_trial_components, 774
list_trials, 774
list_triggers, 429
list_trust_anchors, 451
list_trust_store_certificates, 947
list_trust_stores, 947
list_trusted_token_issuers, 861
list_type_registrations, 114
list_type_versions, 114
listTyped_link_facet_attributes, 110
listTyped_link_facet_names, 110
list_types, 114
list_unsupported_app_version_resources, 726
list_updates, 347
list_usage_for_license_configuration, 531
list_usage_limits, 716
list_usage_totals, 464
list_use_cases, 239
list_user_access_logging_settings, 947
list_user_associations, 536
list_user_groups, 690
list_user_hierarchy_groups, 239
list_user_import_jobs, 216
list_user_policies, 447
list_user_pool_clients, 216
list_user_pools, 216
list_user_proficiencies, 239
list_user_profiles, 201, 774
list_user_settings, 947
list_user_tags, 447
list_users, 216, 239, 394, 447, 454, 578, 690, 722, 937
list_users_by_permission_group, 394
list_users_in_group, 216
list_variants_import_jobs, 606
list_variant_stores, 606
list_vaults, 419
list_vector_enrichment_jobs, 782
list_verified_email_addresses, 820
list_versions, 561, 612
list_versions_by_function, 515
list_view_versions, 240
list_views, 240, 729
list_virtual_clusters, 374
list_virtual_gateways, 44
list_virtual_interface_test_history, 289
list_virtual_machines, 82
list_virtual_mfa_devices, 447
list_virtual_nodes, 44
list_virtual_routers, 44
list_virtual_services, 44
list_vocabulary_filters, 898
list_volume_initiators, 868
list_volume_recovery_points, 868
list_volumes, 868
list_vpc_association_authorizations, 738
list_vpc_connections, 483, 690
list_vpc_endpoints, 360, 612
list_vpc_endpoint_access, 360, 612
list_vpc_endpoints_for_domain, 360, 612
list_vpc_ingress_connections, 50
list_waiting_workflow_steps, 458
list_watchlists, 907
list_web_ac_ls, 914, 918, 922
list_webhooks, 198
list_website_authorization_providers, 933
list_website_certificate_authorities, 933
list_what_if_analyses, 408
list_what_if_forecast_exports, 408
list_what_if_forecasts, 408
list_work_groups, 61
list_worker_blocks, 581
list_worker_configurations, 486
list_workers_with_qualification_type, 581
list_workflow_build_versions, 458
list_workflow_executions, 458
list_workflow_runs, 173
list_workflow_step_executions, 458
list_workflow_types, 880
list_workflows, 173, 268, 429, 458, 606
list_workforces, 774
list_workgroups, 716
list_workload_shares, 926
list_workloads, 41, 926
list_workspaces, 561, 673
list_workteams, 774
list_xss_match_sets, 914, 918
list_zonal_shifts, 57
locationservice, 543
lock_rule, 705
lock_snapshot, 326
logout, 857
lookoutequipment, 546
lookoutmetrics, 549
lookup_developer_identity, 211
lookup_events, 133
lookup_policy, 110
machinelearning, 552
macie2, 555
manage_propertygraph_statistics, 591
manage_sparql_statistics, 591
managedgrafana, 559
marketplacecatalog, 562
marketplacecommerceanalytics, 565
marketplaceentitlementservice, 567
marketplacemetering, 570
memorydb, 573
merge_branches_by_fast_forward, 180
merge_branches_by_squash, 180
merge_branches_by_three_way, 180
merge_developer_identities, 211
merge_profiles, 268
merge_pull_request_by_fast_forward, 180
merge_pull_request_by_squash, 180
merge_pull_request_by_three_way, 180
merge_shards, 498
meter_usage, 572
migrate_workspace, 943
modify_account, 943
modify_activity_stream, 699
modify_address_attribute, 326
modify_aqua_configuration, 710
modify_authentication_profile, 710
modify_availability_zone_group, 326
modify_backup_attributes, 125
modify_cache_cluster, 351
modify_cache_parameter_group, 351
modify_cache_subnet_group, 351
modify_capacity_reservation, 326
modify_capacity_reservation_fleet, 326
modify_certificate_based_auth_properties, 943
modify_certificates, 699
modify_client_properties, 943
modify_client_vpn_endpoint, 326
modify_cluster, 125, 370, 710
modify_cluster_db_revision, 710
modify_cluster_iam_roles, 710
modify_cluster_maintenance, 710
modify_cluster_parameter_group, 710
modify_cluster_snapshot, 710
modify_cluster_snapshot_schedule, 710
modify_cluster_subnet_group, 710
modify_current_db_cluster_capacity, 699
modify_custom_db_engine_version, 699
modify_custom_domain_association, 710
modify_db_cluster, 298, 587, 699
modify_db_cluster_endpoint, 587, 699
modify_db_cluster_parameter_group, 298, 587, 699
modify_db_cluster_snapshot_attribute, 298, 587, 699
modify_db_instance, 298, 587, 699
modify_db_parameter_group, 587, 699
modify_db_proxy, 699
modify_db_proxy_endpoint, 699
modify_db_proxy_target_group, 699
modify_db_recommendation, 699
modify_db_shard_group, 699
modify_db_snapshot, 699
modify_db_snapshot_attribute, 699
modify_db_subnet_group, 298, 587, 699
modify_default_credit_specification, 326
modify_document_permission, 845
modify_ebs_default_kms_key_id, 326
modify_endpoint_access, 710
modify_event_subscription, 298, 587, 699, 710
modify_fleet, 326
modify_fpga_image_attribute, 326
modify_global_cluster, 298, 587, 699
modify_global_replication_group, 351
modify_hapg, 122
modify_hosts, 326
modify_hsm, 122
modify_id_format, 326
modify_identity_id_format, 326
modify_image_attribute, 326
modify_instance_attribute, 326
modify_instance_capacity_reservation_attributes, 326
modify_instance_credit_specification, 326
modify_instance_event_start_time, 326
modify_instance_event_window, 326
modify_instance_fleet, 371
modify_instance_groups, 371
modify_instance_maintenance_options, 326
modify_instance_metadata_defaults, 326
modify_instance_metadata_options, 326
modify_instance_placement, 326
modify_integration, 699
modify_ipam, 326
modify_ipam_pool, 326
modify_ipam_resource_cidr, 326
modify_ipam_resource_discovery, 326
modify_ipam_scope, 326
modify_launch_template, 326
modify_listener, 367
modify_load_balancer_attributes, 364, 367
modify_local_gateway_route, 326
modify_luna_client, 122
modify_managed_prefix_list, 326
modify_mount_target_security_groups, 343
modify_network_interface_attribute, 326
modify_option_group, 700
modify_private_dns_name_options, 326
modify_redshift_idc_application, 710
modify_replication_group, 351
modify_replication_group_shard_configuration, 351
modify_report_definition, 261
modify_reserved_instances, 326
modify_rule, 367
modify_saml_properties, 943
modify_scheduled_action, 710
modify_security_group_rules, 326
modify_selfservice_permissions, 943
modify_serverless_cache, 351
modify_snapshot_attribute, 326
modify_snapshot_copy_retention_period, 710
modify_snapshot_schedule, 710
modify_snapshot_tier, 326
modify_spot_fleet_request, 326
modify_subnet_attribute, 326
modify_target_group, 367
modify_target_group_attributes, 367
modify_tenant_database, 700
modify_traffic_mirror_filter_network_services, 326
modify_traffic_mirror_filter_rule, 326
modify_traffic_mirror_session, 326
modify_transit_gateway, 326
modify_transit_gateway_prefix_list_reference, 326
modify_transit_gateway_vpc_attachment, 326
modify_trust_store, 367
modify_usage_limit, 710
modify_user, 351
modify_user_group, 351
modify_verified_access_endpoint, 327
modify_verified_access_endpoint_policy, 327
modify_verified_access_group, 327
modify_verified_access_group_policy, 327
modify_verified_access_instance, 327
modify_verified_access_instance_logging_configuration, 327
modify_verified_access_trust_provider, 327
modify_volume, 327
modify_volume_attribute, 327
modify_vpc_attribute, 327
modify_vpc_endpoint, 327
modify_vpc_endpoint_connection_notification, 327
modify_vpc_endpoint_service_configuration, 327
modify_vpc_endpoint_service_payer_responsibility, 327
modify_vpc_endpoint_service_permissions, 327
modify_vpc_peering_connection_options, 327
modify_vpc_tenancy, 327
modify_vpn_connection, 327
modify_vpn_connection_options, 327
modify_vpn_tunnel_certificate, 327
modify_vpn_tunnel_options, 327
modify_workspace_access_properties, 943
modify_workspace_creation_properties, 943
modify_workspace_properties, 943
modify_workspace_state, 943
monitor_contact, 240
monitor_instances, 327
move_account, 627
move_address_to_vpc, 327
move_byoip_cidr_to_ipam, 327
mq, 576
mturk, 578
mwaa, 581
neptune, 584
neptunedata, 588
networkfirewall, 591
networkmanager, 595
nimblestudio, 599
notify_object_complete, 84
notify_provision_product_engine_workflow_result, 811
notify_recommendations_received, 255
notify_resource_deployment_status_change, 679
notify_terminate_provisioned_product_engine_workflow_result, 811
notify_update_provisioned_product_engine_workflow_result, 811
notify_when_uploaded, 868
notify_workers, 581
omics, 602
open_instance_public_ports, 541
opensearchingestion, 606
opensearchservice, 609
opensearchserviceserverless, 612
opsworks, 616
opsworkscm, 620
opt_in_phone_number, 837
organizations, 624
override_pull_request_approval_rules, 181

panorama, 628
pause_campaign, 244
pause_cluster, 710
pause_contact, 240
pause_service, 50
paymentcryptographycontrolplane, 631
paymentcryptographydataplane, 634
pcaconnectorad, 637
peer_vpc, 542
personalize, 639
personalizeevents, 643
personalizeruntime, 645
phone_number_validate, 654
pi, 648
pinpoint, 651
pinpointemail, 655
pinpointmsvoice, 659
pinpointmsvoicev2, 661
poll_for_activity_task, 880
poll_for_decision_task, 880
poll_for_jobs, 198
poll_for_task, 271
poll_for_third_party_jobs, 198
polly, 666
post_agent_profile, 188
post_comment_for_compared_commit, 181
post_comment_for_pull_request, 181
post_comment_reply, 181
post_content, 526
post_text, 526
post_time_series_data_points, 275
post_to_connection, 25
predict, 555
prepare_query, 892
preview_agents, 461
pricing, 668
prometheusservice, 671
promote, 578
promote_permission_created_from_policy, 694
promote_read_replica, 700
promote_read_replica_db_cluster, 587, 700
promote_resource_share_created_from_policy, 694
proton, 674
provide_anomaly_feedback, 264
provision_byoip_cidr, 327, 423
provision_device, 630
provision_ipam_byoasn, 327
provision_ipam_pool_cidr, 327
provision_permission_set, 861
provision_product, 811
provision_public_ipv_4_pool_cidr, 327
publish, 837
publish_app_version, 726
publish_batch, 837
publish_function, 119
publish_layer_version, 515
publish_metrics, 584
publish_package_version, 165
publish_recipe, 433
publish_schema, 110
publish_state_machine_version, 828
publish_type, 114
publish_version, 515
purchase_capacity_block, 327
purchase_host_reservation, 327
purchase_provisioned_capacity, 419
purchase_reserved_cache_nodes_offering, 351
purchase_reserved_db_instances_offering, 700
purchase_reserved_elasticsearch_instance_offering, 360
purchase_reserved_instance_offering, 612
purchase_reserved_instances_offering, 327
purchase_reserved_node_offering, 710
purchase_reserved_nodes_offering, 575
purchase_scheduled_instances, 327
purge_queue, 840
push_domain, 741
put_access_control_rule, 937
put_access_grants_instance_resource_policy, 762
put_access_point_configuration_for_object_lambda, 762
put_access_point_policy, 762
put_access_point_policy_for_object_lambda,
  762
put_account_alias, 878
put_account_configuration, 15
put_account_dedicated_ip_warmup_attributes,
  658, 825
put_account_details, 825
put_account_policy, 153
put_account_preferences, 343
put_account_sending_attributes, 658,
  825
put_account_setting, 340
put_account_setting_default, 340
put_account_suppression_attributes, 825
put_account_vdm_attributes, 825
put_action_interactions, 645
put_action_revision, 198
put_actions, 645
put_admin_account, 403
put_aggregation_authorization, 234
put_alarm, 542
put_alert_manager_definition, 673
put_alternate_contact, 13
put_anomaly_detector, 139
put_application_access_scope, 861
put_application_assignment_configuration,
  861
put_application_authentication_method,
  861
put_application_grant, 861
put_application_policy, 807
put_approval_result, 198
put_apps_list, 403
put_attribute_mapping, 451
put_attributes, 340, 834
put_audit_events, 136
put_auth_policy, 910
put_auto_scaling_policy, 371
put_auto_termination_policy, 371
put_backup_policy, 343
put_backup_vault_access_policy, 79
put_backup_vault_lock_configuration,
  79
put_backup_vault_notifications, 79
put_bandwidth_rate_limit_schedule, 82
put_block_public_access_configuration,
  371
put_bot, 519
put_bot_alias, 519
put_bucket_accelerate_configuration,
  758
put_bucket_acl, 758
put_bucket_analytics_configuration,
  758
put_bucket_cors, 758
put_bucket_encryption, 758
put_bucket_intelligent_tiering_configuration,
  758
put_bucket_inventory_configuration,
  758
put_bucket_lifecycle, 758
put_bucket_lifecycle_configuration,
  758, 762
put_bucket_logging, 758
put_bucket_metrics_configuration, 758
put_bucket_notification, 758
put_bucket_notification_configuration,
  758
put_bucket_ownership_controls, 758
put_bucket_policy, 758, 763
put_bucket_replication, 758, 763
put_bucket_request_payment, 758
put_bucket_tagging, 758, 763
put_bucket_versioning, 758, 763
put_bucket_website, 758
put_capacity_assignment_configuration,
  61
put_case_event_configuration, 247
put_chunk, 84
put_classification_export_configuration,
  558
put_cluster_capacity_providers, 340
put_cluster_policy, 483
put_code_binding, 792
put_comment_reaction, 181
put_compliance_items, 845
put_component_policy, 458
put_composite_alarm, 139
put_config_rule, 234
put_configuration, 47
put_configuration_aggregator, 234
put_configuration_recorder, 234
put_configuration_set_delivery_options,
  658, 820, 825
put_configuration_set_reputation_options, 658, 825
put_configuration_set_sending_options, 658, 825
put_configuration_set_suppression_options, 825
put_configuration_set_tracking_options, 658, 825
put_configuration_set_vdm_options, 825
put_conformance_pack, 234
put_contact_information, 13
put_contact_policy, 849
put_container_recipe_policy, 458
put_core_network_policy, 599
put_dashboard, 139
put_data_catalog_encryption_settings, 429
put_data_lake_settings, 511
put_data_protection_policy, 153, 837
put_data_set_refresh_properties, 690
put_dedicated_ip_in_pool, 658, 825
put_dedicated_ip_pool_scaling_attributes, 825
put_dedicated_ip_warmup_attributes, 658, 825
put_deliverability_dashboard_option, 658, 825
put_delivery_channel, 234
put_delivery_destination, 153
put_delivery_destination_policy, 153
put_delivery_source, 153
put_destination, 153
put_destination_policy, 153
put_detector, 412
put_dial_request_batch, 244
put_domain_permissions_policy, 165
put_draft_app_version_template, 726
put_email_identity_configuration_set_attributes, 825
put_email_identity_dkim_attributes, 658, 825
put_email_identity_dkim_signing_attributes, 825
put_email_identity_feedback_attributes, 658, 825
put_email_identity_mail_from_attributes, 658, 825
put_email_monitoring_configuration,
put_keyword, 665
put_kms_encryption_key, 412
put_label, 412
put_launch_action, 304
put_launch_profile_members, 602
put_lexicon, 668
put_lifecycle_configuration, 343
put_lifecycle_event_hook_execution_status, 185
put_lifecycle_hook, 72
put_lifecycle_policy, 334
put_log_events, 153
put_logging_configuration, 914, 918, 922
put_mailbox_permissions, 937
put_maintenance_start_time, 82
put_managed_insight_rules, 139
put_managed_rule_set_versions, 922
put_managed_scaling_policy, 371
put_metadata, 471
put_method, 23
put_method_response, 23
put_metric_alarm, 139
put_metric_data, 139
put_metric_filter, 153
put_metric_stream, 139
put_mobile_device_access_override, 937
put_model_invocation_logging_configuration, 90
put_model_package_group_policy, 774
put_multi_region_access_point_policy, 763
put_notification_channel, 403
put_notification_configuration, 72
put_notification_settings, 451
put_object, 84, 758
put_object_acl, 758
put_object_legal_hold, 758
put_object_lock_configuration, 758
put_object_retention, 758
put_object_tagging, 759
put_opted_out_number, 665
put_organization_config_rule, 234
put_organization_conformance_pack, 234
put_outcome, 412
put_package_origin_configuration, 165
put_parameter, 845
put_partner_events, 143, 383
put_permission, 143, 188, 383
put_permission_policy, 914, 918, 922
put_permissions_boundary_to_permission_set, 861
put_pipeline_definition, 271
put_policy, 18, 380, 403
put_principal_mapping, 489
put_profile_object, 268
put_profile_object_type, 268
put_project_events, 146
put_project_policy, 722
put_protocols_list, 403
put_provisioned_concurrency_config, 515
put_public_access_block, 759, 763
put_query_definition, 153
put_raw_message_content, 939
put_recommendation_feedback, 191
put_recommendation_preferences, 230
put_record, 397, 498, 780
put_record_batch, 397
put_records, 498
put_registration_field_value, 665
put_registry_catalog_data, 337
put_registry_policy, 334
put_registry_scanning_configuration, 334
put_remediation_configurations, 234
put_remediation_exceptions, 234
put_replication_configuration, 334
put_report_definition, 38, 261
put_repository_catalog_data, 337
put_repository_permissions_policy, 165
put_repository_triggers, 181
put_resolver_query_log_policy, 754
put_resource_config, 403
put_resource_permission, 854
put_resource_policy, 133, 153, 168, 223, 307, 429, 498, 549, 565, 595, 599, 627, 710, 716, 792, 795, 845, 852, 910, 950
put_resource_set, 403
put_rest_api, 23
put_restore_validation_result, 79
put_retention_configuration, 234
put_retention_policy, 153, 937
put_role_permissions_boundary, 447
put_role_policy, 447
put_rule, 143, 383
put_rule_groups_namespace, 673
put_rum_events, 159
put_rum_metrics_destination, 159
put_runtime_management_config, 35, 72
put_scheduled_action, 35
put_scheduled_update_group_action, 72
put_schema, 904
put_schema_from_json, 110
put_schema_version_metadata, 429
put_secret_value, 795
put_service_quota_increase_request_into_template, 817
put_session, 526, 528
put_sink_policy, 156
put_slot_type, 519
put_snapshot_block, 313
put_sol_function_package_content, 886
put_sol_network_package_content, 886
put_storage_lens_configuration, 763
put_storage_lens_configuration_tagging, 763
put_stored_query, 234
put_studio_members, 602
put_subscription_filter, 153
put_suppressed_destination, 825
put_targets, 143, 383
put_telemetry_records, 950
put_third_party_job_failure_result, 198
put_third_party_job_success_result, 198
put_trace_segments, 950
put_user_permissions_boundary, 447
put_user_policy, 447
put_user_status, 240
put_users, 643
put_warm_pool, 72
put_webhook, 198
put_workflow_run_properties, 429

qldb, 680
qldbSession, 682
query, 307, 489, 892
query_assistant, 255
query_forecast, 405
query_lineage, 774
query_objects, 271
query_schema_version_metadata, 429
query_what_if_forecast, 405
quicksight, 685
ram, 691
rds, 694
rdsdataservice, 701
re_encrypt, 508
re_encrypt_data, 636
rebalance_slots_in_global_replication_group, 351
reboot_broker, 483, 578
reboot_cache_cluster, 351
reboot_cluster, 710
reboot_db_cluster, 700
reboot_db_instance, 298, 587, 700
reboot_db_shard_group, 700
reboot_instance, 542, 619
reboot_instances, 327
reboot_node, 278
reboot_relational_database, 542
reboot_workspaces, 943
rebuild_environment, 354
rebuild_workspaces, 943
receive_message, 840
recognize_celebrities, 722
recognize_text, 528
recognize_utterance, 528
record_activity_task_heartbeat, 881
record_handler_progress, 114
record_lifecycle_action_heartbeat, 72
recyclebin, 703
redrive_execution, 828
redshift, 706
redshiftdataapiservice, 711
redshiftserverless, 714
refresh_cache, 868
refresh_trusted_advisor_check, 875
register_account, 66
register_trusted_advisor_check, 881
register_application, 854
register_application_revision, 185
register_certificate, 292
register_client, 864
register_cluster, 347
register_container_image, 542
register_container_instance, 340
register_cross_account_access_role, 461
register_data_lake_delegated_administrator, 804
register_db_proxy_targets, 700
register_default_patch_baseline, 845
register_delegated_administrator, 627
register_device, 220
register_devices, 774
register_domain, 741, 881
register_ecs_cluster, 619
register_elastic_ip, 619
register_event_topic, 292
register_identity_provider, 536
register_image, 327
register_instance, 619, 814
register_instance_event_notification_attributes, 327
register_instances_with_load_balancer, 364
register_job_definition, 87
register_mail_domain, 937
register_on_premises_instance, 185
register_organization_admin_account, 66
register_organization_delegated_admin, 134
register_package_version, 630
register_patch_baseline_for_patch_group, 845
register_publisher, 114
register_rds_db_instance, 619
register_resource, 511
register_scalable_target, 35
register_schema_version, 429
register_slack_workspace_for_organization, 878
register_stream_consumer, 498
register_target_with_maintenance_window, 845
register_targets, 367, 910
register_task_definition, 340
register_task_with_maintenance_window, 845
register_to_work_mail, 937
register_transit_gateway, 599
register_transit_gateway_multicast_group_members, 327
register_transit_gateway_multicast_group_sources, 327
register_type, 114
register_usage, 572
register_user, 690
register_volume, 619
register_webhook_with_third_party, 198
register_workflow_type, 881
register_workspace_directory, 943
reimport_api, 29
reject_account_link_invitation, 943
reject_assignment, 581
reject_attachment, 599
reject_client_vpc_connection, 483
reject_data_share, 710
reject_domain_transfer_from_another_aws_account, 741
reject_environment_account_connection, 679
reject_grant, 531
reject_inbound_connection, 612
reject_inbound_cross_cluster_search_connection, 361
reject_invitation, 282
reject_portfolio_share, 811
reject_predictions, 275
reject_qualification_request, 581
reject_resource_share_invitation, 694
reject_shared_directory, 292
reject_subscription_request, 275
reject_transit_gateway_multicast_domain_associations, 327
reject_transit_gateway_peering_attachment, 327
reject_transit_gateway_vpc_attachment, 327
reject_vpc_endpoint_connections, 327
reject_vpc_peering_connection, 327
rekognition, 717
release_address, 327
release_file_system_nfs_v3_locks, 415
release_hosts, 327
release_ipam_pool_allocation, 327
release_phone_number, 240, 665
release_sender_id, 665
release_static_ip, 542
remove_account_from_organization, 627
remove_all_resource_permissions, 930
remove_application_instance, 630
remove_attributes, 654
remove_attributes_from_findings, 461
remove_auto_scaling_policy, 371
remove_auto_termination_policy, 371
remove_client_id_from_open_id_connect_provider, 447
remove_custom_routing_endpoints, 423
remove_draft_app_version_resource_mappings, 726
remove_endpoints, 423
remove_facet_from_object, 110
remove_from_global_cluster, 298, 587, 700
remove_ip_routes, 292
remove_knowledge_base_template_uri, 255
remove_layer_version_permission, 515
remove lf_tags_from_resource, 511
remove_listener_certificates, 367
remove_managed_scaling_policy, 371
remove_notification_channel, 188, 285
remove_permission, 143, 188, 383, 516, 837, 840
remove_region, 292
remove_regions_from_replication, 795
remove_resource_permission, 930
remove_role_from_db_cluster, 587, 700
remove_role_from_db_instance, 700
remove_role_from_instance_profile, 447
remove_schema_version_metadata, 429
remove_source_identifier_from_subscription, 298, 587, 700
remove_tags, 134, 271, 361, 364, 367, 371, 612
remove_tags_from_certificate, 15
remove_tags_from_on_premises_instances, 185
remove_tags_from_resource, 122, 292, 298, 351, 588, 700, 845, 868
remove_tags_from_stream, 498
remove_tags_from_vault, 419
remove_targets, 143, 383
remove_trust_store_revocations, 367
remove_user_from_group, 447
remove_workload, 41
render_ui_template, 774
renew_certificate, 15
renew_domain, 741
reorder_receipt_rule_set, 820
replace_iam_instance_profile_association, 327
replace_network_acl_association, 327
replace_network_acl_entry, 327
replace_permission_associations, 694
replace_route, 328
replace_route_table_association, 328
replace_transit_gateway_route, 328
replace_vpn_tunnel, 328
replicate_instance, 240
replicate_key, 508
replicate_secret_to_regions, 796
report_instance_status, 328
report_task_progress, 271
report_task_runner_heartbeat, 271
request_cancel_workflow_execution, 881
request_certificate, 15
request_environment_info, 354
request_phone_number, 665
request_sender_id, 665
request_service_quota_increase, 817
request_spot_fleet, 328
request_spot_instances, 328
rescore, 492
resend_confirmation_code, 216
resend_contact_reachability_email, 741
resend_operation_authorization, 741
resend_validation_email, 15
reset_address_attribute, 328
reset_authorizers_cache, 29
reset_cache, 868
reset_cache_parameter_group, 351
reset_cluster_parameter_group, 710
reset_db_cluster_parameter_group, 298, 588, 700
reset_db_parameter_group, 588, 700
reset_distribution_cache, 542
reset_ebs_default_kms_key_id, 328
reset_enabled_baseline, 259
reset_encryption_key, 464
reset_fpga_image_attribute, 328
reset_image_attribute, 328
reset_instance_attribute, 328
reset_job_bookmark, 429
reset_landing_zone, 259
reset_network_interface_attribute, 328
reset_notification_settings, 451
reset_parameter_group, 575
reset_password, 937
reset_service_setting, 845
reset_service_specific_credential, 447
reset_snapshot_attribute, 328
reset_user_password, 292, 394
resiliencehub, 723
resize_cluster, 710
resolve_app_version_resources, 726
resolve_case, 875
resolve_customer, 572
resourceexplorer, 727
resourcegroups, 730
resourcegroupstaggingapi, 733
respond_activity_task_canceled, 881
respond_activity_task_completed, 881
respond_activity_task_failed, 881
respond_decision_task_completed, 881
respond_to_auth_challenge, 216
restart_app_server, 354
restore_address_to_classic, 328
restore_analysis, 690
restore_backup, 125
restore_certificate_authority, 18
restore_cluster_from_snapshot, 301
restore_core_network_policy_version, 599
restore_db_cluster_from_s3, 700
restore_db_cluster_from_snapshot, 298, 588, 700
restore_db_cluster_to_point_in_time, 298, 588, 700
restore_db_instance_from_db_snapshot, 700
restore_db_instance_from_s3, 700
restore_db_instance_to_point_in_time, 700
restore_document_versions, 930
restore_domain_access, 933
restore_event_data_store, 134
restore_from_cluster_snapshot, 711
restore_from_recovery_point, 716
restore_from_snapshot, 292, 716
restore_image_from_recycle_bin, 328
restore_key, 634
restore_managed_prefix_list_version, 328
restore_object, 759
restore_secret, 796
restore_server, 623
restore_snapshot_from_recycle_bin, 328
restore_snapshot_tier, 328
restore_table, 495
restore_table_from_backup, 307
restore_table_from_cluster_snapshot, 711
restore_table_from_recovery_point, 717
restore_table_from_snapshot, 717
restore_table_to_point_in_time, 307
restore_volume_from_snapshot, 415
restore_workspace, 943
resume_batch_load_task, 894
resume_campaign, 244
resume_cluster, 711
resume_contact, 240
resume_contact_recording, 240
resume_processes, 72
resume_resource, 408
resume_service, 50
resume_session, 845
resume_workflow_run, 429
resync_mfa_device, 448
retire_grant, 508
retrieve, 489
retrieve_domain_auth_code, 741
retrieve_environment_info, 354
retrieve_tape_archive, 868
retrieve_tape_recovery_point, 868
retry_build, 168
retry_build_batch, 169
retry_data_replication, 304
retry_pipeline_execution, 774
retry_stage_execution, 198
return_savings_plan, 790
reverse_replication, 304
revoke_cache_security_group_ingress, 351
revoke_certificate, 18
revoke_client_vpn_ingress, 328
revoke_cluster_security_group_ingress, 711
revoke_db_security_group_ingress, 700
revoke_domain_access, 933
revoke_endpoint_access, 711
revoke_grant, 508
revoke_ip_rules, 943
revoke_permissions, 511
revoke_security_group_egress, 328
revoke_security_group_ingress, 328
revoke_snapshot_access, 711
revoke_subscription, 275
revoke_token, 216
revoke_vpc_endpoint_access, 361, 612
rollback_application, 503
rollback_instance_refresh, 72
rollback_stack, 114
rollback_transaction, 703
rotate_encryption_key, 711
rotate_key_on_demand, 508
rotate_secret, 796
route53, 735
route53domains, 739
route53recoverycluster, 742
route53recoverycontrolconfig, 745
route53recoveryreadiness, 748
route53resolver, 751
run_instances, 328
run_job_flow, 371
run_scheduled_instances, 328
run_statement, 429
run_task, 340
s3, 755
s3control, 759
s3outposts, 763
sagemaker, 766
sagemakeredgemanager, 776
sagemakerfeaturestoreruntime, 778
sagemakergeospatialcapabilities, 780
sagemakermetrics, 783
sagemakerruntime, 785
savingsplans, 788
scan, 308
scan_provisioned_products, 811
schedule_key_deletion, 508
schemas, 790
search, 130, 275, 729, 774
search_analyses, 690
search_associated_transcripts, 523
search_available_phone_numbers, 240
search_cases, 247
search_contacts, 240
search_content, 255
search_dashboards, 690
search_data_sets, 690
search_data_sources, 690
search_databases_by_lf_tags, 511
search_devices, 98
search_faces, 722
search_faces_by_image, 722
search_folders, 690
search_group_profiles, 275
search_groups, 690
search_hours_of_operations, 240
search_insights, 285
search_jobs, 98
search_listings, 275
search_local_gateway_routes, 328
search_organization_insights, 285
search_place_index_for_position, 545
search_place_index_for_suggestions, 545
search_place_index_for_text, 545
search_predefined_attributes, 240
search_products, 811
search_products_as_admin, 811
search_profiles, 268
search_prompts, 240
search_provisioned_products, 811
search_quantum_tasks, 98
search_queues, 240
search_quick_connects, 240
search_quick_responses, 255
search_raster_data_collection, 782
search_related_items, 247
search_resource_tags, 240
search_resources, 558, 732, 930
search_routing_profiles, 240
search_schemas, 792
search_security_profiles, 240
search_sessions, 255
search_tables, 429
search_tables_by_lf_tags, 511
search_transit_gateway_multicast_groups, 328
search_transit_gateway_routes, 328
search_types, 275
search_user_profiles, 275
search_users, 240, 722
search_users_by_image, 722
search_vocabulary, 240
search_vulnerabilities, 464
secretsmanager, 793
securityhub, 796
securitylake, 801
select, 834
select_aggregate_resource_config, 234
select_resource_config, 234
send_activation_code, 849
send_aggregation_report, 234
send_object_content, 759
send_resource_config, 234
send_activation_code, 849
send_aggregation_report, 234
send_object_content, 759
send_resource_config, 234
send_account_default_protect_configuration, 665
send_account_default_protect_configuration, 665
send_automation_signal, 845
send_bonus, 581
send_bounce, 820
send_bulk_email, 825
send_chat_integration_event, 240
send_cis_session_health, 464
send_cis_session_telemetry, 464
send_command, 684, 845
send_contact_method_verification, 542
send_custom_verification_email, 821
send_destination_number_verification_code, 665
send_diagnostic_interrupt, 328
send_email, 659, 821, 825
send_event, 252, 412, 476
send_heartbeat, 777
send_media_message, 665
send_message, 252, 840
send_message_batch, 840
send_messages, 654
send_otp_message, 654
send_pipeline_execution_step_failure, 774
send_pipeline_execution_step_success, 774
send_project_session_action, 433
send_raw_email, 821
send_serial_console_ssh_public_key, 331
send_ssh_public_key, 331
send_task_failure, 828
send_task_heartbeat, 828
send_task_success, 828
send_test_email, 821
send_test_event_notification, 581
send_test_message, 665
send_users_messages, 654
send_voice_message, 661, 665
send_workflow_step_action, 458
serverlessapplicationrepository, 805
servicecatalog, 808
servicediscovery, 812
servicequotas, 815
ses, 818
sesv2, 821
set_account_default_protect_configuration, 665
set_active_receipt_rule_set, 821
set_alarm_state, 139
set_cognito_events, 220
set_data_retrieval_policy, 419
set_default_message_type, 665
set_default_permission_version, 694
set_default_policy_version, 448
set_default_sender_id, 665
set_desired_capacity, 72
set_endpoint_attributes, 837
set_identity_dkim_enabled, 821
set_identity_feedback_forwarding_enabled, 821
set_identity_headers_in_notifications_enabled, 821
set_identity_mail_from_domain, 821
set_identity_notification_topic, 221
set_identity_pool_configuration, 220
set_identity_pool_roles, 211
set_instance_health, 72
set_instance_protection, 72
set_ip_address_type, 367, 542
set_job_flow_termination_policy, 774
set_job_flow_termination_policy, 774
set_log_delivery_configuration, 216
set_media_message_spend_limit_override, 665
set_permission, 619
set_platform_application_attributes, 837
INDEX

set_principal_tag_attribute_map, 211
set_queue_attributes, 840
set_receipt_rule_position, 821
set_repository_policy, 334, 337
set_resource_access_for_bucket, 542
set_risk_configuration, 216
set_rule_priorities, 367
set_repository_policy, 334, 337
set_resource_access_for_bucket, 542
set_risk_configuration, 216
set_rule_priorities, 367
set_security_groups, 367
set_security_token_service_preferences, 448
set_smb_guest_password, 868
set_sms_attributes, 837
set_stack_policy, 114
set_status, 271
set_subnets, 367
set_subscription_attributes, 837
set_tags_for_resource, 461
set_task_status, 271
set_termination_protection, 371
set_text_message_spend_limit_override, 665
set_time_based_auto_scaling, 620
set_topic_attributes, 837
set_type_configuration, 114
set_type_default_version, 114
set_ui_customization, 216
set_unhealthy_node_replacement, 371
set_user_mfa_preference, 216
set_user_pool_mfa_config, 216
set_user_settings, 216
set_vault_access_policy, 419
set_vault_notifications, 419
set_visible_to_all_users, 371
set_voice_message_spend_limit_override, 665
setup_instance_https, 542
sfn, 826
share_directory, 292
shield, 829
shutdown_gateway, 868
sign, 508
sign_out_user, 933
sign_up, 216
signal_application_instance_node_instances, 630
signal_resource, 114
signal_workflow_execution, 881
simpledb, 832
simulate_custom_policy, 448
simulate_principal_policy, 448
skip_wait_time_for_instance_termination, 185
sns, 834
split_shard, 498
sqs, 838
ssm, 841
ssmcontacts, 846
ssmincidents, 849
ssmsap, 852
sso, 855
ssoadmin, 857
ssooidc, 861
start_activity_stream, 700
start_annotation_import_job, 606
start_app_assessment, 726
start_app_block_builder, 54
start_application, 376, 501, 503
start_application_refresh, 854
start_assessment_framework_share, 66
start_assessment_run, 461
start_asset_bundle_export_job, 690
start_asset_bundle_import_job, 690
start_associations_once, 845
start_attachment_upload, 252
start_automation_execution, 845
start_availability_monitor_test, 868
start_backup_job, 79
start_bgp_failover_test, 289
start_blueprint_run, 429
start_bot_recommendation, 523
start_bot_resource_generation, 523
start_build, 169
start_build_batch, 169
start_calculation_execution, 61
start_call_analytics_job, 898
start_campaign, 244
start_canary, 884
start_celebrity_recognition, 722
start_change_request_execution, 845
start_change_set, 565
start_chat_contact, 240
start_cis_session, 464
start_cluster, 301
start_column_statistics_task_run, 429
start_composition, 480
start_config_rules_evaluation, 234
start_media_analysis_job, 722
start_medical_scribe_job, 898
start_medical_transcription_job, 898
start_message_move_task, 840
start_metadata_generation_run, 275
start_metric_streams, 139
start_migration, 351, 519
start_misconfigured_state_recovery, 415
start_ml_data_processing_job, 591
start_ml_evaluation_task_run, 429
start_ml_labeling_set_generation_task_run, 429
start_ml_model_training_job, 591
start_ml_model_transform_job, 591
start_monitoring_member, 282
start_monitoring_members, 437
start_monitoring_schedule, 774
start_network_insights_access_scope_analysis, 328
start_network_insights_analysis, 328
start_notebook_execution, 371
start_notebook_instance, 774
start_object, 84
start_organization_service_access_update, 599
start_outbound_voice_contact, 240
start_person_tracking, 722
start_phi_detection_job, 226
start_pii_entities_detection_job, 223
start_pipe, 386
start_pipeline, 608
start_pipeline_execution, 198, 774
start_policy_generation, 10
start_product_subscription, 536
start_project_session, 433
start_project_version, 722
start_query, 134, 149, 153
start_query_execution, 61
start_query_planning, 511
start_read_set_activation_job, 606
start_read_set_export_job, 606
start_read_set_import_job, 606
start_recommender, 643
start_recovery, 304
start_reference_import_job, 606
start_remediation_execution, 234
start_replay, 143, 383
start_replication, 304
start_report_creation, 735
start_report_job, 79
start_resource_evaluation, 234
start_resource_scan, 10, 114
start_resource_state_update, 458
start_restore_job, 79
start_retraining_scheduler, 549
start_route_analysis, 599
start_run, 606
start_rx_norm_inference_job, 226
start_savings_plans_purchase_recommendation_generation, 265
start_schema_extension, 292
start_segment_detection, 722
start_sentiment_detection_job, 223
start_service_software_update, 612
start_session, 61, 845
start_snapshot, 313
start_snomedct_inference_job, 227
start_source_network_recovery, 304
start_source_network_replication, 304
start_speaker_enrollment_job, 907
start_speech_synthesis_task, 668
start_stack, 620
start_stream_encryption, 498
start_stream_processor, 722
start_streaming_session, 602
start_studio_sso_configuration_repair, 602
start_support_data_export, 567
start_sync_execution, 828
start_targeted_sentiment_detection_job, 223
start_test_execution, 523
start_test_set_generation, 523
start_text_detection, 723
start_text_translation_job, 900
start_topics_detection_job, 223
start_transaction, 511
start_transcription_job, 898
start_trigger, 429
start_user_access_tasks, 32
start_user_import_job, 216
start_variant_import_job, 606
start_vector_enrichment_job, 783
start_viewer_session_revocation, 471
start_virtual_machines_metadata_sync, 82
start_vpc_endpoint_service_private_dns_verification, 328
start_web_rtc_contact, 240
start_workflow_execution, 881
start_workflow_run, 173, 429
start_workspaces, 943
start_zonal_shift, 57
stop_activity_stream, 700
stop_app_block_builder, 54
stop_application, 376, 501, 503
stop_assessment_run, 461
stop_auto_ml_job, 774
stop_automation_execution, 845
stop_backup_job, 79
stop_bgp_failover_test, 289
stop_bot_recommendation, 523
stop_build, 169
stop_build_batch, 169
stop_calculation_execution, 61
stop_campaign, 244
stop_canary, 884
stop_cis_session, 465
stop_cluster, 301
stop_column_statistics_task_run, 429
stop Compilation_job, 774
stop_composition, 480
stop_configuration_recorder, 234
stop_contact, 240
stop_contact_recording, 240
stop_contact_streaming, 240
stop_crawler, 429
stop_crawler_schedule, 429
stop_data_source_sync_job, 489
stop_db_cluster, 298, 588, 700
stop_db_instance, 700
stop_db_instance_automated_backups_replication, 700
stop_delivery_stream_encryption, 397
stop_deployment, 185
stop_dev_environment, 173
stop_dev_environment_session, 173
stop_discoverer, 792
stop dominant_language_detection_job, 223
stop earth_observation_job, 783
stop_edge_deployment_stage, 774
stop_edge_packaging_job, 774
stop_employment, 849
stop_entities_detection_job, 223
stop_entities_detection_v2_job, 227
stop_evaluation_job, 90
stop_event_data_store_ingestion, 134
stop_events_detection_job, 223
stop_execution, 828
stop_experiment, 146, 399
stop_fallback, 304
stop_fleet, 54
stop_gui_session, 542
stop_human_loop, 68
stop_hyper_parameter_tuning_job, 774
stop_icd10cm_inference_job, 227
stop_image_builder, 54
stop_import, 134
stop_inference_experiment, 774
stop_inference_recommendations_job, 774
stop_inference_scheduler, 549
stop_ingestion, 32
stop_instance, 542, 620
stop_instances, 328
stop_job_run, 433
stop_key_phrases_detection_job, 224
stop_key_usage, 634
stop_labeling_job, 774
stop_launch, 146
stop_logging, 134
stop_metric_streams, 139
stop_model_customization_job, 90
stop_monitoring_members, 437
stop_monitoring_schedule, 774
stop_notebook_execution, 371
stop_notebook_instance, 774
stop_phi_detection_job, 227
stop pii_entities_detection_job, 224
stop_pipeline, 386
stop_pipeline_execution, 198, 774
stop_processing_job, 774
stop_product_subscription, 536
stop_project_version, 723
stop_query, 149, 153
stop_query_execution, 61
update_annotation_store, 606
update_annotation_store_version, 606
update_anomaly, 153
update_anomaly_detector, 552
update_anomaly_monitor, 265
update_anomaly_subscription, 265
update_answer, 926
update_api, 29
update_api_destination, 143, 383
update_api_key, 23
update_api_mapping, 29
update_apns_channel, 654
update_apns_sandbox_channel, 654
update_apns_voip_sandbox_channel, 654
update_apns_voip_channel, 654
update_app, 620, 726
update_app_authorization, 32
update_app_block_builder, 54
update_app_image_config, 774
update_app_monitor, 159
update_app_version, 726
update_app_version_app_component, 726
update_app_version_resource, 726
update_application, 41, 47, 54, 185, 355, 377, 501, 504, 807, 861
update_application_layer_automatic_response, 831
update_application_maintenance_configuration, 831
update_application_resource_lifecycle, 355
update_application_settings, 654, 854
update_application_version, 355
update_approval_rule_template_content, 181
update_approval_rule_template_description, 181
update_approval_rule_template_name, 181
update_archive, 143, 383
update_archive_rule, 11
update_artifact, 774
update_assessment, 66
update_assessment_control, 66
update_assessment_control_set_status, 66
update_assessment_framework, 66
update_assessment_framework_share, 66
untag_resources, 735
untag_role, 448
untag_saml_provider, 448
untag_user, 448
update_accelerator, 423
update_accelerator_attributes, 423
update_access_control_configuration, 490
update_access_entry, 347
update_access_grants_location, 763
update_access_key, 448
update_access_log_subscription, 910
update_access_policy, 615
update_account, 23
update_account_configuration, 193
update_account_customization, 690
update_account_password_policy, 448
update_account_sending_enabled, 821
update_account_settings, 615, 679, 690, 732, 892
update_acl, 575
update_action, 774
update_action_target, 801
update_action_type, 198
update_active_model_version, 549
update_adapter, 889
update_addon, 347
update_adm_channel, 654
update_agent_status, 240
update_alert, 552
update_alias, 508, 516, 634
update_allow_list, 558
update_analysis, 690
update_analysis_permissions, 690
untag_resources, 735
untag_role, 448
untag_saml_provider, 448
untag_user, 448
update_accelerator, 423
update_accelerator_attributes, 423
update_access_control_configuration, 490
update_access_entry, 347
update_access_grants_location, 763
update_access_key, 448
update_access_log_subscription, 910
update_access_policy, 615
update_account, 23
update_account_configuration, 193
update_account_customization, 690
update_account_password_policy, 448
update_account_sending_enabled, 821
update_account_settings, 615, 679, 690, 732, 892
update_acl, 575
update_action, 774
update_action_target, 801
update_action_type, 198
update_active_model_version, 549
update_adapter, 889
update_addon, 347
update_adm_channel, 654
update_agent_status, 240
update_alert, 552
update_alias, 508, 516, 634
update_allow_list, 558
update_analysis, 690
update_analysis_permissions, 690
update_assessment_status, 66
update_assessment_target, 461
update_association, 845
update_assumption_status, 845
update_attribute_group, 47
update_audit_stream_configuration, 933
update_auth_event_feedback, 216
update_authorizer, 23, 29
update_auto_scaling_group, 72
update_automated_discovery_group, 558
update_automated_discovery_configuration, 558
update_automatic_tape_creation_policy, 868
update_availability_configuration, 937
update_availability_options, 128
update_backup_plan, 79
update_baidu_channel, 654
update_bandwidth_rate_limit, 868
update_bandwidth_rate_limit_schedule, 868
update_base_path_mapping, 23
update_batch_prediction, 555
update_billing_group, 95
update_blueprint, 429
update_bot, 523
update_bot_alias, 523
update_bot_locale, 523
update_bot_recommendation, 523
update_broker, 578
update_broker_count, 483
update_broker_storage, 483
update_broker_type, 483
update_broker_settings, 483
update_bucket, 542
update_bucket_bundle, 542
update_budget, 101
update_budget_action, 101
update_byte_match_set, 914, 918
update_cache_policy, 119
update_calculated_attribute_definition, 268
update_call_analytics_category, 898
update_campaign, 643, 655
update_campaign_dialer_config, 244
update_campaign_name, 244
update_campaign_outbound_call_config, 244
update_canary, 884
update_capacity_provider, 340
update_capacity_reservation, 61
update_case, 247
update_cell, 750
update_certificate_authority, 18
update_certificate_options, 15
update_changeset, 394
update_channel, 134, 471
update_chap_credentials, 868
update_cis_scan_configuration, 465
update_classification_job, 558
update_classification_scope, 558
update_classifier, 430
update_client_certificate, 23
update_cloud_front_origin_access_identity, 119
update_cluster, 278, 301, 340, 575, 774
update_cluster_config, 347
update_cluster_configuration, 483
update_cluster_kafka_version, 483
update_cluster_settings, 340
update_cluster_software, 774
update_cluster_version, 347
update_code_repository, 775
update_code_signing_config, 516
update_collection, 615
update_column_statistics_for_partition, 430
update_column_statistics_for_table, 430
update_comment, 181
update_company_network_configuration, 933
update_component, 41, 679
update_component_configuration, 41
update_compute_environment, 87
update_conditional_forwarder, 292
update_configuration, 465, 484, 578
update_configuration_policy, 801
update_configuration_set_event_destination, 659, 661, 821, 825
update_configuration_set_reputation_metrics_enabled, 821
update_configuration_set_sending_enabled, 821
update_configuration_set_tracking_options, 821
update_configuration_template, 355
update_connect_client_add_in, 943
update_connection, 143, 289, 383, 430, 599
update_connection_alias_permission, 943
update_connectivity, 484
update_connector, 486
update_constraint, 811
update_contact, 240, 825, 849
update_contact_attributes, 240
update_contact_channel, 849
update_contact_evaluation, 240
update_contact_flow_content, 240
update_contact_flow_metadata, 240
update_contact_flow_module_content, 240
update_contact_flow_module_metadata, 240
update_contact_flow_name, 240
update_contact_list, 825
update_contact_routing_data, 241
update_contact_schedule, 241
update_container_agent, 340
update_container_instances_state, 340
update_container_service, 542
update_context, 255
update_continent, 775
update_continuous_backups, 308
update_continuous_deployment_policy, 119
update_contributor_insights, 308
update_control, 66
update_control_panel, 747
update_core_network, 599
update_cost_allocation_tags_status, 265
update_cost_category_definition, 265
update_crawler, 430
update_crawler_schedule, 430
update_crl, 431
update_cross_account_attachment, 423
update_custom_domain_association, 717
update_custom_key_store, 508
update_custom_line_item, 95
update_custom_routing_accelerator, 423
update_custom_routing_accelerator_attributes, 423
update_custom_routing_listener, 423
update_custom_verification_email_template, 821, 825
update_dashboard, 690
update_dashboard_links, 691
update_dashboard_permissions, 691
update_dashboard_published_version, 691
update_data_catalog, 61
update_data_cells_filter, 511
update_data_lake, 804
update_data_lake_exception_subscription, 804
update_data_quality_ruleset, 430
update_data_repository_association, 416
update_data_set, 691
update_data_set_permissions, 691
update_data_source, 275, 490, 555, 612, 691
update_data_source_permissions, 691
update_database, 430, 895
update_dataset, 394, 433, 643
update_dataset_entries, 723
update_dataset_group, 409
update_datasource_packages, 282
update_default_auto_scaling_configuration, 50
update_default_branch, 181
update_default_mail_domain, 937
update_deletion_protection, 852
update_deployment, 23, 29
update_deployment_group, 185
update_destination, 397
update_detector, 437
update_detector_version, 412
update_detector_version_metadata, 412
update_detector_version_status, 412
update_dev_endpoint, 412
update_dev_environment, 173
update_device, 599
update_device_fleet, 775
update_device_metadata, 631
update_device_policy_configuration, 933
update_device_status, 216
update_device_status, 775
update_direct_connect_gateway, 289
update_direct_connect_gateway_association,
update_firewall_rule, 754
update_firewall_rule_group_association, 754
update_fleet, 54, 169
update_fleet_metadata, 933
update_flywheel, 224
update_folder, 691, 930
update_folder_permissions, 691
update_framework, 79
update_function, 119
update_function_code, 516
update_function_configuration, 516
update_function_event_invoke_config, 516
update_function_url_config, 516
update_gateway_information, 82, 868
update_gateway_response, 23
update_gateway_route, 44
update_gcm_channel, 655
update_generated_template, 114
update_geo_match_set, 914, 918
update_geofence_collection, 546
update_global_network, 599
update_global_settings, 79, 926
update_global_table, 308
update_global_table_settings, 308
update_glossary, 275
update_glossary_term, 275
update_group, 216, 448, 454, 691, 732, 937, 950
update_group_profile, 275
update_group_query, 732
update_guardrail, 90
update_health_check, 738
update_hit_review_status, 581
update_hit_type_of_hit, 581
update_host, 205
update_hosted_zone_comment, 738
update_hours_of_operation, 241
update_http_namespace, 814
update_hub, 775
update_hypervisor, 82
update_hypervisor_policy_assignment, 691
update_id_mapping_workflow, 380
update_id_namespace, 380
update_identity_pool, 211
update_identity_propagation_config, 691
update_identity_provider, 216, 947
update_identity_provider_configuration, 933
update_identity_provider_settings, 536
update_identity_source, 904
update_image, 775
update_image_permissions, 54
update_image_pipeline, 458
update_image_version, 775
update_impersonation_role, 937
update_in_app_template, 655
update_incident_record, 852
update_index, 490
update_index_type, 729
update_inference_component, 775
update_inference_component_runtime_config, 775
update_inference_experiment, 775
update_inference_scheduler, 549
update_infrastructure_configuration, 458
update_ingestion_destination, 32
update_insight, 801
update_instance, 620, 861
update_instance_access_control_attribute_configuration, 861
update_instance_attribute, 241
update_instance_custom_health_status, 814
update_instance_metadata_options, 542
update_instance_storage_config, 241
update_integration, 23, 29, 926
update_integration_response, 23, 29
update_intent, 523
update_investigation_state, 282
update_ip_access_settings, 947
update_ip_restriction, 691
update_ip_set, 437, 914, 918, 923
update_item, 308
update_job, 430
update_job_from_source_control, 430
update_job_priority, 763
update_job_queue, 87
update_job_status, 763
update_journey, 655
update_journey_state, 655
update_key, 546
update_notification_rule, 208
update_notification_settings, 581
update_number_of_domain_controllers, 293
update_object_attributes, 110
update_open_id_connect_provider_thumbprint, 448
update_ops_item, 846
update_ops_metadata, 846
update_org_ec2_deep_inspection_configuration, 465
update_organization_configuration, 282, 437, 465, 558, 801
update_organizational_unit, 628
update_origin_access_control, 119
update_origin_request_policy, 119
update_outpost_resolver, 754
update_package, 361, 612
update_package_group, 165
update_package_group_origin_configuration, 165
update_package_versions_status, 165
update_parallel_data, 901
update_parameter_group, 278, 575
update_participant_role_config, 241
update_partition, 430
update_partner_status, 711
update_patch_baseline, 846
update_permission_group, 394
update_permission_set, 861
update_permissions, 561
update_phone_number, 241, 665
update_phone_number_metadata, 241
update_pipe, 386
update_pipeline, 198, 608, 775
update_pipeline_execution, 775
update_place_index, 546
update_playback_restriction_policy, 471
update_pod_identity_association, 347
update_policy, 628, 904
update_policy_store, 904
update_policy_template, 904
update_pool, 665
update_portal, 947
update_portfolio, 811
update_portfolio_share, 811
update_practice_run_configuration, 57
update_predefined_attribute, 241
update_prepared_statement, 241
update_pricing_plan, 95
update_pricing_rule, 95
update_primary_email_address, 937
update_primary_region, 508
update_private_dns_namespace, 814
update_problem, 41
update_product, 811
update_profile, 268, 451, 926
update_profile_job, 433
update_profiling_group, 188
update_project, 146, 169, 173, 202, 275, 433, 775
update_project_data_delivery, 146
update_project_visibility, 169
update_prompt, 241
update_protect_configuration, 665
update_protect_configuration_country_rule_set, 665
update_protection_group, 831
update_provisioned_model_throughput, 90
update_provisioned_product, 811
update_provisioned_product_properties, 811
update_provisioning_artifact, 811
update_public_dns_namespace, 814
update_public_key, 119
update_public_sharing_settings, 691
update_publishing_destination, 437
update_pull_request_approval_rule_content, 181
update_pull_request_approval_state, 181
update_pull_request_description, 181
update_pull_request_status, 181
update_pull_request_title, 181
update_pull_through_cache_rule, 334
update_push_template, 655
update_qualification_type, 581
update_query_suggestions_block_list, 490
update_query_suggestions_config, 490
update_queue_hours_of_operation, 241
update_queue_max_contacts, 241
update_queue_name, 241
update_queue_outbound_caller_config,
update_queue_status, 241
update_quick_connect_config, 241
update_quick_connect_name, 241
update_quick_response, 255
update_radius, 293
update_rate_based_rule, 914, 918
update_rds_db_instance, 620
update_readiness_check, 750
update_realtime_log_config, 119
update_receipt_rule, 821
update_recipe, 433
update_recipe_job, 433
update_recommender, 643
update_recommender_configuration, 655
update_records, 220
update_recovery_group, 750
update_recovery_point_lifecycle, 79
update_refresh_schedule, 691
update_regex_match_set, 914, 918
update_regex_pattern_set, 914, 918, 923
update_region_settings, 79
update_registry, 430, 792
update_related_items, 852
update_relational_database, 542
update_relational_database_parameters,
update_replication_configuration, 304
update_replication_info, 484
update_replication_set, 852
update_report_definition, 38
update_report_group, 169
update_report_plan, 79
update_repository, 165
update_repository_description, 181
update_repository_encryption_key, 181
update_repository_link, 205
update_repository_name, 181
update_request_validator, 23
update_rescore_execution_plan, 492
update_resiliency_policy, 726
update_resolver_config, 754
update_resolver_dnssec_config, 754
update_resolver_endpoint, 754
update_resolver_rule, 754
update_resource, 23, 107, 511, 937
update_resource_collection, 285
update_resource_data_sync, 846
update_resource_policy, 523
update_resource_profile, 558
update_resource_profile_detections,
update_resource_server, 216
update_resource_set, 750
update_resource_share, 694
update_response_headers_policy, 119
update_response_plan, 852
update_rest_api, 23
update_restore_testing_plan, 79
update_restore_testing_selection, 79
update_retraining_scheduler, 549
update_reveal_configuration, 558
update_review_template, 926
update_review_template_answer, 926
update_review_template_lens_review,
update_role, 448
update_role_custom_permission, 691
update_role_description, 448
update_room, 476
update_rotation, 849
update_route, 29, 44
update_route_calculator, 546
update_route_response, 29
update_routing_control, 747
update_routing_control_state, 745
update_routing_profile_agent_availability_timer,
update_routing_profile_concurrency,
update_routing_profile_default_outbound_queue,
update_routing_profile_name, 241
update_routing_profile_queues, 241
update_rule, 241, 705, 910, 914, 918
update_rule_group, 595, 914, 918, 923
update_rule_metadata, 413
update_rule_version, 413
update_rules_of_ip_group, 943
update_ruleset, 433
update_rum_metric_definition, 159
update_run_group, 606
update_safety_rule, 747
update_saml_provider, 448
update_sampling_rule, 950
update_scaling_parameters, 128
update_scaling_plan, 75
update_schedule, 388, 433
update_scheduled_action, 612, 717
update_scheduled_query, 892
update_scheduling_policy, 87
update_schema, 110, 430, 792
update_schema_mapping, 380
update_secret, 796
update_secret_version_stage, 796
update_security, 484
update_security_config, 615
update_security_control, 801
update_security_group_rule_descriptions_egress, 328
update_security_group_rule_descriptions_ingress, 328
update_security_hub_configuration, 801
update_security_policy, 615
update_security_profile, 241
update_segment, 655
update_sender_id, 665
update_sensitivity_inspection_template, 558
update_server, 623
update_server_certificate, 448
update_server_engine_attributes, 623
update_service, 50, 340, 679, 814, 910
update_service_access_policies, 128
update_service_action, 812
update_service_instance, 679
update_service_integration, 285
update_service_network, 910
update_service_network_vpc_association, 910
update_service_pipeline, 679
update_service_primary_task_set, 340
update_service_setting, 846
update_service_settings, 532, 534
update_service_specific_credential, 448
update_service_sync_blocker, 679
update_service_sync_config, 679
update_service_template, 679
update_service_template_version, 679
update_settings, 66, 293
update_shard_count, 498
update_share_invitation, 926
update_shared_vpc_configuration, 416
update_signing_certificate, 448
update_site, 599
update_size_constraint_set, 914, 918
update_slack_channel_configuration, 878
update_slot, 523
update_slot_type, 523
update_smb_file_share, 868
update_smb_file_share_visibility, 868
update_smb_local_groups, 868
update_smb_security_strategy, 868
update_sms_channel, 655
update_sms_template, 655
update_snapshot, 416, 717
update_snapshot_schedule, 869
update_sol_function_package, 887
update_sol_network_instance, 887
update_sol_network_package, 887
update_source_control_from_job, 430
update_space, 173, 775
update_spice_capacity_configuration, 691
update_sql_injection_match_set, 914, 918
update_ssh_public_key, 448
update_stack, 54, 114, 620
update_stack_instances, 114
update_stack_set, 115
update_stage, 23, 29, 480
update_standards_control, 801
update_state_machine, 828
update_state_machine_alias, 828
update_storage, 484
update_storage_lens_group, 763
update_storage_virtual_machine, 416
update_stream_mode, 498
update_stream_processor, 723
update_streaming_distribution, 119
update_streaming_image, 602
update_studio, 371, 602
update_studio_component, 602
update_studio_session_mapping, 371
update_subnet_change_protection, 595
update_subnet_group, 278, 575
update_subscriber, 101, 804
update_subscriber_notification, 804
update_subscription, 831
update_subscription_grant_status, 275
update_subscription_request, 276
update_subscription_target, 276
update_sync_blocker, 205
update_sync_configuration, 205
update_table, 308, 430, 495, 895
update_table_objects, 511
update_table_optimizer, 430
update_table_replica_auto_scaling, 308
update_table_storage_optimizer, 511
update_tag_option, 812
update_tags_for_domain, 742
update_tags_for_resource, 355
update_target_account_configuration, 400
update_target_group, 910
update_task_protection, 340
update_task_set, 340
update_task_template, 241
update_team_member, 202
update_template, 247, 639, 691, 821
update_template_active_version, 655
update_template_alias, 691
update_template_group_access_control_entry, 639
update_template_permissions, 691
update_template_sync_config, 679
update_termination_protection, 115
update_test_set, 523
update_theme, 691
update_theme_alias, 691
update_theme_permissions, 691
update_thesaurus, 490
update_threat_intel_set, 437
update_time_to_live, 308
update_timeline_event, 832
update_tls_inspection_configuration, 595
update_topic, 691
update_topic_permissions, 691
update_topic_refresh_schedule, 691
update_tracker, 546
update_traffic_distribution, 241
update_traffic_policy_comment, 739
update_traffic_policy_instance, 739
update_trial, 134
update_training_job, 775
update_trial, 775
update_trial_component, 775
update_trigger, 430
update_trust, 293
update_trust_anchor, 451
update_trust_store, 947
update_trusted_tokenIssuer, 861
update_typed_link_facet, 110
update_usage, 23
update_usage_limit, 717
update_usage_plan, 23
update_user, 394, 448, 454, 575, 578, 691, 930, 937
update_user_access_logging_settings, 947
update_user_attributes, 216
update_user_defined_function, 430
update_user_hierarchy, 241
update_user_hierarchy_group_name, 241
update_user_hierarchy_structure, 241
update_user_identity_info, 241
update_user_phone_config, 241
update_user_pool, 216
update_user_pool_client, 216
update_user_pool_domain, 216
update_user_proficiencies, 241
update_user_profile, 202, 276, 620, 775
update_user_routing_profile, 241
update_user_security_profiles, 241
update_user_settings, 947
update_variable, 413
update_variant_store, 606
update_view, 729
update_view_content, 241
update_view_metadata, 241
update_virtual_gateway, 44
update_virtual_interface_attributes, 289
update_virtual_node, 44
update_virtual_router, 44
update_virtual_service, 44
update_vocabulary, 898
update_vocabulary_filter, 898
update_voice_channel, 655
update_voice_template, 655
update_volume, 416, 620
update_vpc_attachment, 599
update_vpc_connection, 691
update_vpc_endpoint, 361, 612, 615
update_vpc_ingress_connection, 50
update_vpc_link, 23, 29
update_vtl_device_type, 869
update_watchlist, 907
update_web_acl, 914, 918, 923
update_webhook, 169
update_work_group, 61
update_workflow, 430, 606
update_workforce, 61
update_workgroup, 775
update_workload, 41, 926
update_workload_share, 926
update_workspace, 561
update_workspace_alias, 673
update_workspace_authentication, 561
update_workspace_bundle, 943
update_workspace_configuration, 561
update_workspace_image_permission, 943
update_workteam, 775
update_xss_match_set, 914, 918
update_zonal_autoshift_configuration, 57
update_zonal_shift, 57
upgrade_applied_schema, 110
upgrade_domain, 612
upgrade_elasticsearch_domain, 361
upgrade_lens_review, 926
upgrade_profile_version, 926
upgrade_published_schema, 110
upgrade_review_template_lens_review, 926
upload_archive, 419
upload_documents, 130
upload_layer_part, 334, 337
upload_multipart_part, 419
upload_part, 759
upload_part_copy, 759
upload_read_set_part, 606
upload_server_certificate, 448
upload_signing_certificate, 448
upload_ssh_public_key, 448
validate_assessment_report_integrity, 66
validate_configuration_settings, 355
validate_pipeline, 608
validate_pipeline_definition, 271
validate_policy, 11
validate_pull_through_cache_rule, 334
validate_resource_policy, 796
validate_sol_function_package_content, 887
validate_sol_network_package_content, 887
validate_state_machine_definition, 829
validate_template, 115
verifiedpermissions, 901
verify, 508
verify_auth_request_cryptogram, 636
verify_card_validation_data, 636
verify_destination_number, 665
verify_domain_dkim, 821
verify_domain_identity, 821
verify_email_address, 821
verify_email_identity, 821
verify_mac, 508, 636
verify_otp_message, 655
verify_pin_data, 636
verify_session, 173
verify_sms_sandbox_phone_number, 837
verify_software_token, 216
verify_trust, 293
verify_user_attribute, 216
view_billing, 742
voiceid, 905
vpclattice, 907
waf, 911
wafregional, 915
wafv2, 919
wellarchitected, 923
withdraw_byoip_cidr, 328, 423
workdocs, 927
worklink, 930
workmail, 933
workmailmessageflow, 937
workspaces, 940
workspacesweb, 944
write_get_object_response, 759
write_records, 895
xray, 947