Package ‘aws.ecx’

October 12, 2022

Title Communicating with AWS EC2 and ECS using AWS REST APIs

Version 1.0.5

Description Providing the functions for communicating with Amazon Web Services(AWS) Elastic Compute Cloud(EC2) and Elastic Container Service(ECS). The functions will have the prefix 'ecs_' or 'ec2_' depending on the class of the API. The request will be sent via the REST API and the parameters are given by the function argument. The credentials can be set via 'aws_set_credentials'. The EC2 documentation can be found at <https://docs.aws.amazon.com/AWSEC2/latest/APIReference/Welcome.html> and ECS can be found at <https://docs.aws.amazon.com/AmazonECS/latest/APIReference/Welcome.html>.

License GPL-3

Encoding UTF-8

RoxygenNote 7.1.1

Imports methods, rjson, aws.signature, httr, xml2, utils


Suggests knitr, rmarkdown, testthat

VignetteBuilder knitr

NeedsCompilation no

Author Jiefei Wang [aut, cre], Martin Morgan [aut]

Maintainer Jiefei Wang <szwjf08@gmail.com>

Repository CRAN

Date/Publication 2022-01-26 13:02:45 UTC

R topics documented:

aws_set_credentials ........................................................ 12
aws_set_retry_time ......................................................... 14
CommonDoc ................................................................. 15
ec2_accept_reserved_instances_exchange_quote .................. 15
ec2_accept_transit_gateway_multicast_domain_associations .... 17
ec2_accept_transit_gateway_peering_attachment ................. 18
ec2_accept_transit_gateway_vpc_attachment ...................... 20
ec2_accept_vpc_endpoint_connections ............................. 21
ec2_accept_vpc_peering_connection ................................ 22
ec2_advertise_byoip_cidr ............................................. 23
ec2_allocate_address .................................................. 24
ec2_allocate_hosts ..................................................... 27
ec2_apply_security_groups_to_client_vpn_target_network ...... 29
ec2_assign_ipv6_addresses .......................................... 31
ec2_assign_private_ip_addresses .................................. 32
ec2_associate_address ................................................. 34
ec2_associate_client_vpn_target_network .......................... 36
ec2_associate_dhcp_options ........................................ 37
ec2_associate_enclave_certificate_iam_role ...................... 38
ec2_associate_iam_instance_profile ................................ 40
ec2_associate_route_table ........................................... 41
ec2_associate_subnet_cidr_block .................................. 42
ec2_associate_transit_gateway_multicast_domain ............... 43
ec2_associate_transit_gateway_route_table ....................... 45
ec2_associate_vpc_cidr_block ....................................... 46
ec2_attach_classic_link_vpc ........................................ 48
ec2_attach_internet_gateway ....................................... 50
ec2_attach_network_interface ..................................... 51
ec2_attach_volume ..................................................... 53
ec2_attach_vpn_gateway .............................................. 54
ec2_authorize_client_vpn_ingress .................................. 55
ec2_authorize_security_group_egress .............................. 57
ec2_authorize_security_group_ingress ............................. 59
ec2_bundle_instance .................................................. 62
ec2_cancel_bundle_task .............................................. 64
ec2_cancel_capacity_reservation .................................. 65
ec2_cancel_conversion_task ......................................... 66
ec2_cancel_export_task .............................................. 67
ec2_cancel_import_task .............................................. 68
ec2_cancel_reserved_instances_listing ............................ 69
ec2_cancelSpotFleetRequests ....................................... 70
ec2_cancel_spot_instance_requests ................................ 72
ec2_confirm_product_instance ...................................... 73
ec2_copy_fpga_image ................................................ 74
ec2_copy_image ........................................................ 76
ec2_copy_snapshot .................................................... 79
ec2_create_capacity_reservation ................................... 82
ec2_create_carrier_gateway ........................................ 85
ec2_create_vpc_endpoint ............................................ 87
ec2_create_client_vpn_route ....................................... 91
topics documented:

ec2_create_customer_gateway ........................................ 93
ec2_create_default_subnet ........................................... 95
ec2_create_default_vpc ............................................. 96
ec2_create_dhcp_options ............................................ 97
ec2_create_egress_only_internet_gateway ......................... 98
ec2_create_fleet .................................................... 100
ec2_create_flow_logs ............................................... 103
ec2_create_fpga_image .............................................. 106
ec2_create_image ................................................... 108
ec2_create_instance_export_task ................................... 110
ec2_create_internet_gateway ....................................... 112
ec2_create_key_pair ................................................ 113
ec2_create_launch_template ........................................ 114
ec2_create_launch_template_version ................................ 116
ec2_create_local_gateway_route ................................... 118
ec2_create_local_gateway_route_table_vpc_association ........ 119
ec2_create_managed_prefix_list ................................... 121
ec2_create_nat_gateway ........................................... 123
ec2_create_network_acl ............................................. 125
ec2_create_network_acl_entry ..................................... 126
ec2_create_network_insights_path ................................ 128
ec2_create_network_interface ..................................... 131
ec2_create_network_interface_permission ....................... 133
ec2_create_placement_group ....................................... 135
ec2_create_reserved_instances_listing ........................... 137
ec2_create_route ................................................... 138
ec2_create_route_table ............................................. 141
ec2_create_security_group ......................................... 142
ec2_create_snapshot ............................................... 144
ec2_create_snapshots .............................................. 146
ec2_create_spot_datafeed_subscription ......................... 148
ec2_create_subnet .................................................. 149
ec2_create_tags ..................................................... 151
ec2_create_traffic_mirror_filter ................................ 153
ec2_create_traffic_mirror_filter_rule ......................... 154
ec2_create_traffic_mirror_session ............................... 157
ec2_create_traffic_mirror_target ................................ 159
ec2_create_transit_gateway ....................................... 161
ec2_create_transit_gateway_connect .............................. 163
ec2_create_transit_gateway_connect_peer ....................... 164
ec2_create_transit_gateway_multicast_domain .................. 166
ec2_create_transit_gateway_peering_attachment ............... 168
ec2_create_transit_gateway_prefix_list_reference ............ 169
ec2_create_transit_gateway_route ................................ 171
ec2_create_transit_gateway_route_table ....................... 173
ec2_create_transit_gateway_vpc_attachment .................... 174
ec2_create_volume ................................................. 176
ec2_create_vpc ................................................... 179
topics documented:

- `ec2_create_vpc_endpoint` ................................................................. 182
- `ec2_create_vpc_endpoint_connection_notification` ................................. 185
- `ec2_create_vpc_endpoint_service_configuration` ..................................... 186
- `ec2_create_vpeering_connection` ....................................................... 188
- `ec2_create_vpn_connection` .............................................................. 190
- `ec2_create_vpn_connection_route` ...................................................... 192
- `ec2_create_vpn_gateway` .................................................................. 193
- `ec2_delete_carrier_gateway` .............................................................. 195
- `ec2_delete_client_vpn_endpoint` ......................................................... 196
- `ec2_delete_client_vpn_route` .............................................................. 197
- `ec2_delete_customer_gateway` ........................................................... 199
- `ec2_delete_dhcp_options` .................................................................. 200
- `ec2_delete_egress_only_internet_gateway` ............................................ 201
- `ec2_delete_fleets` ............................................................................. 202
- `ec2_delete_flow_logs` ...................................................................... 204
- `ec2_delete_fpga_image` ..................................................................... 205
- `ec2_delete_internet_gateway` .............................................................. 206
- `ec2_delete_key_pair` ......................................................................... 207
- `ec2_delete_launch_template` ............................................................... 208
- `ec2_delete_launch_template_versions` ................................................ 209
- `ec2_delete_local_gateway_route` ......................................................... 210
- `ec2_delete_local_gateway_route_table_vpc_association` ....................... 211
- `ec2_delete_managed_prefix_list` ......................................................... 212
- `ec2_delete_nat_gateway` .................................................................. 213
- `ec2_delete_network_acl` .................................................................... 214
- `ec2_delete_network_acl_entry` ............................................................ 215
- `ec2_delete_network_insights_analysis` ................................................. 216
- `ec2_delete_network_insights_path` ....................................................... 217
- `ec2_delete_network_interface` ............................................................ 218
- `ec2_delete_network_interface_permission` .......................................... 219
- `ec2_delete_placement_group` ............................................................... 220
- `ec2_delete_queued_reserved_instances` .............................................. 221
- `ec2_delete_route` ............................................................................. 222
- `ec2_delete_route_table` .................................................................... 223
- `ec2_delete_security_group` ................................................................. 224
- `ec2_delete_snapshot` ....................................................................... 225
- `ec2_delete_spot_datafeed_subscription` ............................................. 226
- `ec2_delete_subnet` ........................................................................... 227
- `ec2_delete_tags` ............................................................................. 228
- `ec2_delete_traffic_mirror_filter` ....................................................... 229
- `ec2_delete_traffic_mirror_filter_rule` ................................................ 230
- `ec2_delete_traffic_mirror_session` .................................................... 231
- `ec2_delete_traffic_mirror_target` ....................................................... 232
- `ec2_delete_transit_gateway` ............................................................... 233
- `ec2_delete_transit_gateway_connect` .................................................. 234
- `ec2_delete_transit_gateway_connect_peer` .......................................... 235
- `ec2_delete_transit_gateway_multicast_domain` ..................................... 236
- `ec2_delete_transit_gateway_peering_attachment` ................................. 237
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ec2_delete_transit_gateway_prefix_list_reference</td>
<td>246</td>
</tr>
<tr>
<td>ec2_delete_transit_gateway_route</td>
<td>247</td>
</tr>
<tr>
<td>ec2_delete_transit_gateway_route_table</td>
<td>248</td>
</tr>
<tr>
<td>ec2_delete_transit_gateway_vpc_attachment</td>
<td>250</td>
</tr>
<tr>
<td>ec2_delete_volume</td>
<td>251</td>
</tr>
<tr>
<td>ec2_delete_vpc</td>
<td>252</td>
</tr>
<tr>
<td>ec2_delete_vpc_endpoints</td>
<td>253</td>
</tr>
<tr>
<td>ec2_delete_vpc_endpoint_connection_notifications</td>
<td>254</td>
</tr>
<tr>
<td>ec2_delete_vpc_endpoint_service_configurations</td>
<td>255</td>
</tr>
<tr>
<td>ec2_delete_vpc_peering_connection</td>
<td>257</td>
</tr>
<tr>
<td>ec2_delete_vpn</td>
<td>258</td>
</tr>
<tr>
<td>ec2_delete_vpn_connection_route</td>
<td>259</td>
</tr>
<tr>
<td>ec2_delete_vpn_gateway</td>
<td>260</td>
</tr>
<tr>
<td>ec2_deprovision_byoip_cidr</td>
<td>261</td>
</tr>
<tr>
<td>ec2_deregister_image</td>
<td>263</td>
</tr>
<tr>
<td>ec2_deregister_instance_event_notification_attributes</td>
<td>264</td>
</tr>
<tr>
<td>ec2_deregister_transit_gateway_multicast_group_members</td>
<td>265</td>
</tr>
<tr>
<td>ec2_deregister_transit_gateway_multicast_group_sources</td>
<td>266</td>
</tr>
<tr>
<td>ec2_describe_account_attributes</td>
<td>268</td>
</tr>
<tr>
<td>ec2_describe_addresses</td>
<td>269</td>
</tr>
<tr>
<td>ec2_describe_addresses_attribute</td>
<td>271</td>
</tr>
<tr>
<td>ec2_describe_aggregate_id_format</td>
<td>272</td>
</tr>
<tr>
<td>ec2_describe_availability_zones</td>
<td>273</td>
</tr>
<tr>
<td>ec2_describe_bundle_tasks</td>
<td>276</td>
</tr>
<tr>
<td>ec2_describe_byoip_cidrs</td>
<td>277</td>
</tr>
<tr>
<td>ec2_describe_capacity_reservations</td>
<td>278</td>
</tr>
<tr>
<td>ec2_describe_carrier_gateways</td>
<td>281</td>
</tr>
<tr>
<td>ec2_describe_classic_link_instances</td>
<td>283</td>
</tr>
<tr>
<td>ec2_describe_client_vpn_authorization_rules</td>
<td>285</td>
</tr>
<tr>
<td>ec2_describe_client_vpn_connections</td>
<td>286</td>
</tr>
<tr>
<td>ec2_describe_client_vpn_endpoints</td>
<td>288</td>
</tr>
<tr>
<td>ec2_describe_client_vpn_routes</td>
<td>289</td>
</tr>
<tr>
<td>ec2_describe_client_vpn_target_networks</td>
<td>291</td>
</tr>
<tr>
<td>ec2_describe_coip_pools</td>
<td>293</td>
</tr>
<tr>
<td>ec2_describe_conversion_tasks</td>
<td>294</td>
</tr>
<tr>
<td>ec2_describe_customer_gateways</td>
<td>296</td>
</tr>
<tr>
<td>ec2_describe_dhcp_options</td>
<td>297</td>
</tr>
<tr>
<td>ec2_describe_egress_only_internet_gateways</td>
<td>299</td>
</tr>
<tr>
<td>ec2_describe_elastic_gpus</td>
<td>301</td>
</tr>
<tr>
<td>ec2_describe_export_image_tasks</td>
<td>302</td>
</tr>
<tr>
<td>ec2_describe_export_tasks</td>
<td>304</td>
</tr>
<tr>
<td>ec2_describe_fast_snapshot_restores</td>
<td>305</td>
</tr>
<tr>
<td>ec2_describe_fleets</td>
<td>306</td>
</tr>
<tr>
<td>ec2_describe_fleet_history</td>
<td>308</td>
</tr>
<tr>
<td>ec2_describe_fleet_instances</td>
<td>310</td>
</tr>
<tr>
<td>ec2_describe_flow_logs</td>
<td>311</td>
</tr>
<tr>
<td>ec2_describe_fpga_images</td>
<td>313</td>
</tr>
<tr>
<td>ec2_describe_fpga_image_attribute</td>
<td>315</td>
</tr>
</tbody>
</table>
topics documented:

ec2_describe_hosts .................................................. 316
ec2_describe_host_reservations ................................. 318
ec2_describe_host_reservation_offerings ..................... 320
ec2_describe_iam_instance_profile_associations .......... 322
ec2_describe_id_format ........................................... 323
ec2_describe_images ............................................... 325
ec2_describe_image_attribute .................................. 328
ec2_describe_import_image_tasks .............................. 329
ec2_describe_import_snapshot_tasks ......................... 331
ec2_describe_instances ............................................ 332
ec2_describe_instance_attribute ............................... 337
ec2_describe_instance_credit_specifications ............... 338
ec2_describe_instance_event_notification_attributes .... 340
ec2_describe_instance_status .................................. 341
ec2_describe_instance_types .................................... 343
ec2_describe_instance_type_offerings ....................... 347
ec2_describe_internet_gateways ................................. 348
ec2_describe_ipv6_pools .......................................... 350
ec2_describe_key_pairs ............................................ 352
ec2_describe_launch_template_versions ..................... 355
ec2_describe_local_gateways ..................................... 358
ec2_describe_local_gateway_route_tables .................. 360
ec2_describe_local_gateway_route_table_virtual_interface_group_associations . 361
ec2_describe_local_gateway_route_table_vpc_associations . 363
ec2_describe_local_gateway_virtual_interfaces ............ 365
ec2_describe_local_gateway_virtual_interface_groups .... 366
ec2_describe_managed_prefix_lists ................ .......... 368
ec2_describe_moving_addresses .................................. 370
ec2_describe_nat_gateways ....................................... 371
ec2_describe_network_acls ....................................... 373
ec2_describe_network_insights_analyses ..................... 375
ec2_describe_network_insights_paths ......................... 377
ec2_describe_network_interfaces ............................... 379
ec2_describe_network_interface_attribute ........ .......... 382
ec2_describe_network_interface_permissions .......... .. 383
ec2_describe_placement_groups .................................. 385
ec2_describe_prefix_lists ....................................... 386
ec2_describe_principal_id_format .............................. 388
ec2_describe_public_ipv4_pools ................................ 389
ec2_describe_regions .............................................. 391
ec2_describe_reserved_instances ............................... 392
ec2_describe_reserved_instances_listing.................... 395
ec2_describe_reserved_instances_modifications .......... 396
ec2_describe_reserved_instances_offerings ................ 398
ec2_describe_route_tables ...................................... 401
ec2_describe_scheduled_instances ............................ 404
ec2_describe_scheduled_instance_availability ........................................... 406
ec2_describe_security_groups ................................................................. 408
ec2_describe_security_group_references .................................................. 411
ec2_describe_snapshots ............................................................................ 412
ec2_describe_snapshot_attribute ............................................................... 414
ec2_describe_spot_datafeed_subscription .................................................. 416
ec2_describe_spot_fleet_instances .............................................................. 417
ec2_describe_spot_fleet_requests ............................................................... 418
ec2_describe_spot_fleet_request_history .................................................... 420
ec2_describe_spot_instance_requests .......................................................... 421
ec2_describe_spot_price_history ................................................................. 424
ec2_describe_stale_security_groups ............................................................ 427
ec2_describe_subnets .................................................................................. 428
ec2_describe_tags ...................................................................................... 430
ec2_describe_traffic_mirror_filters ........................................................... 432
ec2_describe_traffic_mirror_sessions ......................................................... 434
ec2_describe_traffic_mirror_targets ........................................................... 435
ec2_describe_transit_gateways ................................................................... 437
ec2_describe_transit_gateway_attachments .................................................. 439
ec2_describe_transit_gateway_connects ......................................................... 441
ec2_describe_transit_gateway_connect_peers ............................................... 443
ec2_describe_transit_gateway_multicast_domains ....................................... 444
ec2_describe_transit_gateway_peering_attachments .................................... 446
ec2_describe_transit_gateway_route_tables ............................................... 448
ec2_describe_transit_gateway_vpc_attachments ......................................... 449
ec2_describe_volumes ................................................................................. 451
ec2_describe_volumes_modifications ............................................................ 453
ec2_describe_volume_attribute ................................................................ 455
ec2_describe_volume_status ...................................................................... 457
ec2_describe_vpcs ..................................................................................... 459
ec2_describe_vpc_attribute ....................................................................... 461
ec2_describe_vpc_classic_link ................................................................... 462
ec2_describe_vpc_classic_link_dns_support ............................................... 464
ec2_describe_vpc_endpoints ..................................................................... 465
ec2_describe_vpc_endpoint_connections ..................................................... 467
ec2_describe_vpc_endpoint_connection_notifications ................................ 468
ec2_describe_vpc_endpoint_services ........................................................... 470
ec2_describe_vpc_endpoint_service_configurations .................................... 472
ec2_describe_vpc_endpoint_service_permissions ....................................... 473
ec2_describe_vpc_peering_connections ....................................................... 475
ec2_describe_vpn_connections .................................................................. 477
ec2_describe_vpn_gateways ...................................................................... 479
ec2_detach_classic_link_vpc ..................................................................... 481
ec2_detach_internet_gateway ..................................................................... 482
ec2_detach_network_interface ................................................................... 483
ec2_detach_volume ..................................................................................... 485
ec2_detach_vpn_gateway ............................................................................ 486
ec2_disable_ebs_encryption_by_default ...................................................... 488
ec2_import_instance ........................................... 554
ec2_import_key_pair ........................................... 555
ec2_import_snapshot ........................................... 557
ec2_import_volume ............................................ 559
ec2_modify_address_attribute ................................. 561
ec2_modify_availability_zone_group ......................... 562
ec2_modify_capacity_reservation ............................ 564
ec2_modify_client_vpn_endpoint ............................... 566
ec2_modify_default_credit_specification .................... 569
ec2_modify_ebs_default_kms_key_id ......................... 570
ec2_modify_fleet .............................................. 572
ec2_modify_fpga_image_attribute ............................. 573
ec2_modify_hosts .............................................. 576
ec2_modify_identity_id_format ............................... 577
ec2_modify_id_format .......................................... 579
ec2_modify_image_attribute .................................... 580
ec2_modify_instance_attribute ............................... 582
ec2_modify_instance_capacity_reservation_attributes .... 586
ec2_modify_instance_credit_specification ................... 587
ec2_modify_instance_event_start_time ....................... 589
ec2_modify_instance_metadata_options ....................... 590
ec2_modify_instance_placement ................................ 592
ec2_modify_launch_template ................................... 594
ec2_modify_managed_prefix_list .............................. 596
ec2_modify_network_interface_attribute .................... 597
ec2_modify_reserved_instances ............................... 599
ec2_modify_snapshot_attribute ............................... 600
ec2_modify_spot_fleet_request ............................... 602
ec2_modify_subnet_attribute .................................. 604
ec2_modify_traffic_mirror_filter_network_services ....... 606
ec2_modify_traffic_mirror_filter_rule ....................... 607
ec2_modify_traffic_mirror_session ........................... 610
ec2_modify_transit_gateway ................................... 612
ec2_modify_transit_gateway_prefix_list_reference ........... 613
ec2_modify_transit_gateway_vpc_attachment .................. 615
ec2_modify_volume ............................................. 617
ec2_modify_volume_attribute ................................... 619
ec2_modify_vpc_attribute ...................................... 620
ec2_modify_vpc_endpoint ...................................... 622
ec2_modify_vpc_endpoint_connection_notification .......... 624
ec2_modify_vpc_endpoint_service_configuration .......... 626
ec2_modify_vpc_endpoint_service_permissions .............. 628
ec2_modify_vpc_peering_connection_options ................. 630
ec2_modify_vpc_tenancy ........................................ 631
ec2_modify_vpn_connection .................................... 632
ec2_modify_vpn_connection_options ......................... 634
ec2_modify_vpn_tunnel_certificate ........................... 636
ec2_modify_vpn_tunnel_options ............................... 637
topics documented:

ec2_monitor_instances .................................................. 639
ec2_move_address_to_vpc .................................................... 640
ec2_provision_byoip_cidr .................................................. 641
ec2_purchase_host_reservation ............................................ 643
ec2_purchase_reserved_instances_offering ................................ 645
ec2_purchase_scheduled_instances ........................................ 647
ec2_reboot_instances ....................................................... 648
ec2_register_image ........................................................ 649
ec2_register_instance_event_notification_attributes ...................... 652
ec2_register_transit_gateway_multicast_group_members ..................... 653
ec2_register_transit_gateway_multicast_group_sources ....................... 655
ec2_reject_transit_gateway_multicast_domain_associations ................. 656
ec2_reject_transit_gateway_peering_attachment ............................ 658
ec2_reject_transit_gateway_vpc_attachment ................................ 659
ec2_reject_vpc_endpoint_connections ..................................... 660
ec2_reject_vpc_peering_connection ........................................ 661
ec2_release_address ........................................................ 663
ec2_release_hosts ............................................................ 664
ec2_replace_iam_instance_profile_association ................................ 665
ec2_replace_network_acl_association ....................................... 666
ec2_replace_network_acl_entry ............................................. 668
ec2_replace_route ............................................................ 670
ec2_replace_route_table_association ....................................... 673
ec2_replace_transit_gateway_route ......................................... 674
ec2_report_instance_status .................................................. 676
ec2_request_spot_fleet ...................................................... 678
ec2_request_spot_instances ................................................... 679
ec2_reset_address_attribute ................................................ 683
ec2_reset_ebs_default_kms_key_id .......................................... 684
ec2_reset_fpga_image_attribute ............................................. 685
ec2_reset_image_attribute ................................................... 687
ec2_reset_instance_attribute ................................................. 688
ec2_reset_network_interface_attribute ..................................... 689
ec2_reset_snapshot_attribute ............................................... 691
ec2_restore_address_to_classic ............................................. 692
ec2_restore_managed_prefix_list_version ................................... 693
ec2_revoke_client_vpn_ingress .............................................. 695
ec2_revoke_security_group_egress ......................................... 696
ec2_revoke_security_group_ingress ......................................... 699
ec2_run_instances ............................................................. 701
ec2_run_scheduled_instances ............................................... 709
ec2_search_local_gateway_routes .......................................... 711
ec2_search_transit_gateway_multicast_groups .............................. 712
ec2_search_transit_gateway_routes ....................................... 714
ec2_send_diagnostic_interrupt ............................................. 716
ec2_start_instances .......................................................... 717
ec2_start_network_insights_analysis ...................................... 718
ec2_start_vpc_endpoint_service_private_dns_verification ................... 720
aws_set_credentials

Set or get AWS credentials. This function will be called by the package when loaded.

Usage

```r
aws_set_credentials(
  key_file = NULL,
  access_key_id = NULL,
  secret_access_key = NULL,
  region = NULL,
  profile = NULL
)
```

aws_get_credentials()

aws_get_access_key_id()

aws_get_secret_access_key()

aws_get_region()

aws_set_access_key_id(access_key_id)

aws_set_secret_access_key(secret_access_key)

aws_set_region(region)

aws_list_regions()
aws_set_credentials

Arguments

key_file  The csv credential file that is downloaded from AWS
access_key_id  An AWS Access Key ID
secret_access_key  An AWS Secret Access Key
region  A character string containing the AWS region for the request. If missing, "us-east-1" is assumed.
profile  A character string specifying which profile to use from the file. By default, the profile named in AWS_PROFILE is used, otherwise the "default" profile is used.

Details

The function aws_set_credentials uses aws.signature::locate_credentials internally to determine your credentials. There are a variety of ways to find the credentials, the most common methods are (sorted by the search order)

1. user-supplied values passed to the function
2. environment variables (AWS_ACCESS_KEY_ID, AWS_SECRET_ACCESS_KEY, AWS_DEFAULT_REGION, and AWS_SESSION_TOKEN)
3. a profile in a local credentials dot file in the current working directory, using the profile specified by AWS_PROFILE
4. a profile in a global credentials dot file in a location set by AWS_SHARED_CREDENTIALS_FILE or defaulting typically to "/.aws/credentials" (or another OS-specific location), using the profile specified by AWS_PROFILE

Value

aws_set_credentials: A list containing credentials(with asterisk) and region.
aws_get_credentials: A list containing credentials(with asterisk) and region.
aws_get_access_key_id: The access key id
aws_get_secret_access_key: The secret access key
aws_get_region: The region
aws_set_access_key_id: NULL
aws_set_secret_access_key: NULL
aws_set_region: The old region
aws_list_regions: A vector of available regions

Examples

## Get your credentials from the environment variables or AWS cli
aws_set_credentials()

## show your current credentials
aws_get_credentials()
aws_set_retry_time  Get or set the package settings

Description

Get or set the package settings

Usage

aws_set_retry_time(x)
aws_set_print_on_error(x)
aws_set_network_timeout(x)
aws_get_retry_time()
aws_get_print_on_error()
aws_get_network_timeout()

Arguments

x  the value to be set. For the timeout setting, the unit is seconds.

Value

Setter: The old value Getter: The current value

Examples

## Set the timeout to 10 seconds
aws_set_network_timeout(10)
## Get the timeout setting
aws_get_network_timeout()

## Turn off print on error
aws_set_print_on_error(FALSE)

## Set the retry times to 5
aws_set_retry_time(5)
Common documents

Description

Common documents

Arguments

- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]
- **simplify**: Logical. Whether to simplify the result and handle nextToken in the response [optional]
- **Filter**: Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below) [optional]
- **nextToken**: Characters. The token for the next page of results [optional]
- **nextToken**: Characters. The token for the next page of results [optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

No return value

**ec2_accept_reserved_instances_exchange_quote**

*Accept Reserved Instances Exchange Quote*

Description

Accepts the Convertible Reserved Instance exchange quote described in the GetReservedInstancesExchangeQuote call.
Usage

ec2_accept_reserved_instances_exchange_quote(
    ReservedInstanceId,
    DryRun = NULL,
    TargetConfiguration = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

ReservedInstanceId
   List. The IDs of the Convertible Reserved Instances to exchange for another Convertible Reserved Instance...

DryRun
   Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

TargetConfiguration
   List. The configuration of the target Convertible Reserved Instance to exchange for your current Convertible...[optional]

simplify
   Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
   Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
   Logical. Whether to show an error message when a network error occurs.

retry_time
   Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
   Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
   Character. The region of the AWS service.

Value

A list object or a character vector

ReservedInstanceId

The IDs of the Convertible Reserved Instances to exchange for another Convertible Reserved Instance of the same or higher value.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

TargetConfiguration

The configuration of the target Convertible Reserved Instance to exchange for your current Convertible Reserved Instances.

---

describe

Accept Transit Gateway Multicast Domain Associations

Description

Accepts a request to associate subnets with a transit gateway multicast domain.

Usage

```r
ec2_accept_transit_gateway_multicast_domainAssociations(
  TransitGatewayMulticastDomainId = NULL,
  TransitGatewayAttachmentId = NULL,
  SubnetIds = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **TransitGatewayMulticastDomainId**: Character. The ID of the transit gateway multicast domain.[optional]
- **TransitGatewayAttachmentId**: Character. The ID of the transit gateway attachment.[optional]
- **SubnetIds**: List. The IDs of the subnets to associate with the transit gateway multicast domain.[optional]
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
- **simplify**: Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error  Logical. Whether to show an error message when a network error occurs.
retry_time    Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region        Character. The region of the AWS service.

Value
A list object or a character vector

TransitGatewayMulticastDomainId
The ID of the transit gateway multicast domain.

TransitGatewayAttachmentId
The ID of the transit gateway attachment.

SubnetIds
The IDs of the subnets to associate with the transit gateway multicast domain.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description
Accepts a transit gateway peering attachment request. The peering attachment must be in the pendingAcceptance state.
Usage

```r
ec2_accept_transit_gateway_peering_attachment(
    TransitGatewayAttachmentId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **TransitGatewayAttachmentId**: Character. The ID of the transit gateway attachment.
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response.
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request.
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encountering a network issue. If the request has been sent `retry_time` times but still not able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

- A list object or a character vector

**TransitGatewayAttachmentId**

The ID of the transit gateway attachment.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
ec2_accept_transit_gateway_vpc_attachment

Accept Transit Gateway Vpc Attachment

Description

Accept Transit Gateway Vpc Attachment

Usage

ec2_accept_transit_gateway_vpc_attachment(
    TransitGatewayAttachmentId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

TransitGatewayAttachmentId

Character. The ID of the attachment.

DryRun

Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

simplify

Logical. Whether to simplify the result and handle nextToken in the response[optional]

others

Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error

Logical. Whether to show an error message when a network error occurs.

retry_time

Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout

Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region

Character. The region of the AWS service.

Value

A list object or a character vector

TransitGatewayAttachmentId

The ID of the attachment.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**Description**

Accepts one or more interface VPC endpoint connection requests to your VPC endpoint service.

**Usage**

```r
ec2_accept_vpc_endpoint_connections(  
  ServiceId,  
  VpcEndpointId,  
  DryRun = NULL,  
  simplify = TRUE,  
  others = list(),  
  print_on_error = aws_get_print_on_error(),  
  retry_time = aws_get_retry_time(),  
  network_timeout = aws_get_network_timeout(),  
  region = aws_get_region()  
)
```

**Arguments**

- **ServiceId**: Character. The ID of the VPC endpoint service.
- **VpcEndpointId**: List. The IDs of one or more interface VPC endpoints.
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]
- **simplify**: Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.
Value

A list object or a character vector

ServiceId

The ID of the VPC endpoint service.

VpcEndpointId

The IDs of one or more interface VPC endpoints.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

dry_run_operation

Accept Vpc Peering Connection

Accept Vpc Peering Connection

Usage

Accept Vpc Peering Connection

dry_run

Arguments

DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]

VpcPeeringConnectionId Character. The ID of the VPC peering connection. You must specify this parameter in the request.[optional]

simplify Logical. Whether to simplify the result and handle `nextToken` in the response[optional]
ec2_advertise_byoip_cidr

others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value
A list object or a character vector

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response isDryRunOperation. Otherwise, it is UnauthorizedOperation.

VpcPeeringConnectionId
The ID of the VPC peering connection. You must specify this parameter in the request.

---

ec2_advertise_byoip_cidr

Advertise Byoip Cidr

Description
Advertise Byoip Cidr

Usage
ec2_advertise_byoip_cidr(
Cidr,
DryRun = NULL,
simplify = TRUE,
others = list(),
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)
Arguments

- **Cidr**
  Character. The address range, in CIDR notation.
- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]
- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.
- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**
  Character. The region of the AWS service.

Value

A list object or a character vector

**Cidr**

The address range, in CIDR notation. This must be the exact range that you provisioned. You can't advertise only a portion of the provisioned range.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**ec2_allocate_address**  Allocate Address

---

**Description**

Allocate Address
ec2_allocate_address

Usage

ec2_allocate_address(
    Domain = NULL,
    Address = NULL,
    PublicIpv4Pool = NULL,
    NetworkBorderGroup = NULL,
    CustomerOwnedIpv4Pool = NULL,
    DryRun = NULL,
    TagSpecification = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

Domain      Character. Indicates whether the Elastic IP address is for use with instances in a VPC or instances in EC2-Classic...[optional]
Address     Character. [EC2-VPC] The Elastic IP address to recover or an IPv4 address from an address pool.[optional]
PublicIpv4Pool Character. The ID of an address pool that you own.[optional]
NetworkBorderGroup Character. A unique set of Availability Zones, Local Zones, or Wavelength Zones from which AWS advertises...[optional]
CustomerOwnedIpv4Pool Character. The ID of a customer-owned address pool.[optional]
DryRun      Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
TagSpecification List. The tags to assign to the Elastic IP address.[optional]
simplify    Logical. Whether to simplify the result and handle nextToken in the response[optional]
others      Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time   Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region       Character. The region of the AWS service.
Value

A list object or a character vector

Domain

Indicates whether the Elastic IP address is for use with instances in a VPC or instances in EC2-Classic.

Default: If the Region supports EC2-Classic, the default is standard. Otherwise, the default is vpc.

Address

[EC2-VPC] The Elastic IP address to recover or an IPv4 address from an address pool.

PublicIpv4Pool

The ID of an address pool that you own. Use this parameter to let Amazon EC2 select an address from the address pool. To specify a specific address from the address pool, use the Address parameter instead.

NetworkBorderGroup

A unique set of Availability Zones, Local Zones, or Wavelength Zones from which AWS advertises IP addresses. Use this parameter to limit the IP address to this location. IP addresses cannot move between network border groups.

Use DescribeAvailabilityZones to view the network border groups.

You cannot use a network border group with EC2 Classic. If you attempt this operation on EC2 classic, you will receive an InvalidParameterCombination error. For more information, see Error Codes.

CustomerOwnedIpv4Pool

The ID of a customer-owned address pool. Use this parameter to let Amazon EC2 select an address from the address pool. Alternatively, specify a specific address from the address pool.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

TagSpecification

The tags to assign to the Elastic IP address.
ec2_allocate_hosts

Allocate Hosts

Description
Allocates a Dedicated Host to your account. At a minimum, specify the supported instance type or instance family, the Availability Zone in which to allocate the host, and the number of hosts to allocate.

Usage

ec2_allocate_hosts(  
   AvailabilityZone,  
   Quantity,  
   AutoPlacement = NULL,  
   ClientToken = NULL,  
   InstanceType = NULL,  
   InstanceFamily = NULL,  
   TagSpecification = NULL,  
   HostRecovery = NULL,  
   simplify = TRUE,  
   others = list(),  
   print_on_error = aws_get_print_on_error(),  
   retry_time = aws_get_retry_time(),  
   network_timeout = aws_get_network_timeout(),  
   region = aws_get_region()  
)

Arguments

AvailabilityZone Character. The Availability Zone in which to allocate the Dedicated Host.
Quantity Integer. The number of Dedicated Hosts to allocate to your account with these parameters.
AutoPlacement Character. Indicates whether the host accepts any untargeted instance launches that match its instance type...[optional]
ClientToken Character. Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.[optional]
InstanceType Character. Specifies the instance type to be supported by the Dedicated Hosts.[optional]
InstanceFamily Character. Specifies the instance family to be supported by the Dedicated Hosts.[optional]
TagSpecification List. The tags to apply to the Dedicated Host during creation.[optional]
HostRecovery Character. Indicates whether to enable or disable host recovery for the Dedicated Host.[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others

Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]

print_on_error

Logical. Whether to show an error message when a network error occurs.

retry_time

Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout

Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region

Character. The region of the AWS service.

Value

A list object or a character vector

AvailabilityZone

The Availability Zone in which to allocate the Dedicated Host.

Quantity

The number of Dedicated Hosts to allocate to your account with these parameters.

AutoPlacement

Indicates whether the host accepts any untargeted instance launches that match its instance type configuration, or if it only accepts Host tenancy instance launches that specify its unique host ID.

For more information, see Understanding auto-placement and affinity in the Amazon EC2 User Guide.

Default: on

ClientToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. For more information, see Ensuring Idempotency.

InstanceType

Specifies the instance type to be supported by the Dedicated Hosts. If you specify an instance type, the Dedicated Hosts support instances of the specified instance type only.

If you want the Dedicated Hosts to support multiple instance types in a specific instance family, omit this parameter and specify InstanceFamily instead. You cannot specify InstanceType and InstanceFamily in the same request.
**InstanceFamily**

Specifies the instance family to be supported by the Dedicated Hosts. If you specify an instance family, the Dedicated Hosts support multiple instance types within that instance family.

If you want the Dedicated Hosts to support a specific instance type only, omit this parameter and specify **InstanceType** instead. You cannot specify **InstanceFamily** and **InstanceType** in the same request.

**TagSpecification**

The tags to apply to the Dedicated Host during creation.

**HostRecovery**

Indicates whether to enable or disable host recovery for the Dedicated Host. Host recovery is disabled by default. For more information, see Host recovery in the Amazon EC2 User Guide.

Default: off

---

**Description**

Applies a security group to the association between the target network and the Client VPN endpoint. This action replaces the existing security groups with the specified security groups.

**Usage**

```r
ec2_apply_security_groups_to_client_vpn_target_network( 
  ClientVpnEndpointId, 
  VpcId, 
  SecurityGroupId, 
  DryRun = NULL, 
  simplify = TRUE, 
  others = list(), 
  print_on_error = aws_get_print_on_error(), 
  retry_time = aws_get_retry_time(), 
  network_timeout = aws_get_network_timeout(), 
  region = aws_get_region() 
)
```
Arguments

- **ClientVpnEndpointId**
  Character. The ID of the Client VPN endpoint.

- **VpcId**
  Character. The ID of the VPC in which the associated target network is located.

- **SecurityGroupId**
  List. The IDs of the security groups to apply to the associated target network.

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]

- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  Character. The region of the AWS service.

Value

A list object or a character vector

**ClientVpnEndpointId**

The ID of the Client VPN endpoint.

**VpcId**

The ID of the VPC in which the associated target network is located.

**SecurityGroupId**

The IDs of the security groups to apply to the associated target network. Up to 5 security groups can be applied to an associated target network.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`. 
Assign IPv6 Addresses

Usage

e2_assign_ipv6_addresses(
    NetworkInterfaceId,
    Ipv6AddressCount = NULL,
    Ipv6Addresses = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

NetworkInterfaceId
    Character. The ID of the network interface.

Ipv6AddressCount
    Integer. The number of additional IPv6 addresses to assign to the network interface.[optional]

Ipv6Addresses
    List. One or more specific IPv6 addresses to be assigned to the network interface.[optional]

simplify
    Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
    Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
    Logical. Whether to show an error message when a network error occurs.

retry_time
    Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
    Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
    Character. The region of the AWS service.

Value

A list object or a character vector
NetworkInterfaceId

The ID of the network interface.

Ipv6AddressCount

The number of additional IPv6 addresses to assign to the network interface. The specified number of IPv6 addresses are assigned in addition to the existing IPv6 addresses that are already assigned to the network interface. Amazon EC2 automatically selects the IPv6 addresses from the subnet range. You can\'t use this option if specifying specific IPv6 addresses.

Ipv6Addresses

One or more specific IPv6 addresses to be assigned to the network interface. You can\'t use this option if you\'re specifying a number of IPv6 addresses.

---

ec2_assign_private_ip_addresses

Assign Private Ip Addresses

Description

Assign Private Ip Addresses

Usage

ec2_assign_private_ip_addresses(
    NetworkInterfaceId,
    AllowReassignment = NULL,
    PrivateIpAddress = NULL,
    SecondaryPrivateIpAddressCount = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

NetworkInterfaceId

Character. The ID of the network interface.

AllowReassignment

Logical. Indicates whether to allow an IP address that is already assigned to another network interface or...[optional]
PrivateIpAddress
List. One or more IP addresses to be assigned as a secondary private IP address to the network interface.[optional]

SecondaryPrivateIpAddressCount
Integer. The number of secondary IP addresses to assign to the network interface.[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value
A list object or a character vector

NetworkInterfaceId
The ID of the network interface.

AllowReassignment
Indicates whether to allow an IP address that is already assigned to another network interface or instance to be reassigned to the specified network interface.

PrivateIpAddress
One or more IP addresses to be assigned as a secondary private IP address to the network interface. You can’t specify this parameter when also specifying a number of secondary IP addresses.

If you don’t specify an IP address, Amazon EC2 automatically selects an IP address within the subnet range.

SecondaryPrivateIpAddressCount
The number of secondary IP addresses to assign to the network interface. You can’t specify this parameter when also specifying private IP addresses.
ec2_associate_address  Associate Address

Description
Associate Address

Usage

```r
ec2_associate_address(
    AllocationId = NULL,
    InstanceId = NULL,
    PublicIp = NULL,
    AllowReassociation = NULL,
    DryRun = NULL,
    NetworkInterfaceId = NULL,
    PrivateIpAddress = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **AllocationId**: Character. [EC2-VPC] The allocation ID. This is required for EC2-VPC.[optional]
- **InstanceId**: Character. The ID of the instance.[optional]
- **PublicIp**: Character. [EC2-Classic] The Elastic IP address to associate with the instance.[optional]
- **AllowReassociation**: Logical. [EC2-VPC] For a VPC in an EC2-Classic account, specify true to allow an Elastic IP address that...[optional]
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
- **NetworkInterfaceId**: Character. [EC2-VPC] The ID of the network interface.[optional]
- **PrivateIpAddress**: Character. [EC2-VPC] The primary or secondary private IP address to associate with the Elastic IP address.[optional]
- **simplify**: Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
**retry_time**  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

**network_timeout**  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

**region**  Character. The region of the AWS service.

**Value**

A list object or a character vector

**AllocationId**

[EC2-VPC] The allocation ID. This is required for EC2-VPC.

**InstanceId**

The ID of the instance. The instance must have exactly one attached network interface. For EC2-VPC, you can specify either the instance ID or the network interface ID, but not both. For EC2-Classic, you must specify an instance ID and the instance must be in the running state.

**PublicIp**

[EC2-Classic] The Elastic IP address to associate with the instance. This is required for EC2-Classic.

**AllowReassociation**

[EC2-VPC] For a VPC in an EC2-Classic account, specify true to allow an Elastic IP address that is already associated with an instance or network interface to be reassigned with the specified instance or network interface. Otherwise, the operation fails. In a VPC in an EC2-VPC-only account, reassociation is automatic, therefore you can specify false to ensure the operation fails if the Elastic IP address is already associated with another resource.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

**NetworkInterfaceId**

[EC2-VPC] The ID of the network interface. If the instance has more than one network interface, you must specify a network interface ID.

For EC2-VPC, you can specify either the instance ID or the network interface ID, but not both.
PrivateIpAddress

[EC2-VPC] The primary or secondary private IP address to associate with the Elastic IP address. If no private IP address is specified, the Elastic IP address is associated with the primary private IP address.

---

**ec2_associate_client_vpn_target_network**

*Associate Client Vpn Target Network*

---

**Description**

Associate Client Vpn Target Network

**Usage**

```r
ec2_associate_client_vpn_target_network(
  ClientVpnEndpointId,
  SubnetId,
  ClientToken = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- **ClientVpnEndpointId**
  Character. The ID of the Client VPN endpoint.

- **SubnetId**
  Character. The ID of the subnet to associate with the Client VPN endpoint.

- **ClientToken**
  Character. Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.[optional]

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request,[optional]

- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
ec2_associate_dhcp_options

network_timeout
   Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
   Character. The region of the AWS service.

Value
   A list object or a character vector

ClientVpnEndpointId
   The ID of the Client VPN endpoint.

SubnetId
   The ID of the subnet to associate with the Client VPN endpoint.

ClientToken
   Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. For more information, see How to Ensure Idempotency.

DryRun
   Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_associate_dhcp_options

Associate Dhcp Options

Description
   Associate Dhcp Options

Usage

ec2_associate_dhcp_options(
   DhcpOptionsId,
   VpcId,
   DryRun = NULL,
   simplify = TRUE,
   others = list(),
   print_on_error = aws_get_print_on_error(),
   retry_time = aws_get_retry_time(),
   network_timeout = aws_get_network_timeout(),
   region = aws_get_region()
)
**Arguments**

- **DhcpOptionsId**  Character. The ID of the DHCP options set, or default to associate no DHCP options with the VPC.
- **VpcId**  Character. The ID of the VPC.
- **DryRun**  Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]
- **simplify**  Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others**  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**  Logical. Whether to show an error message when a network error occurs.
- **retry_time**  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- **network_timeout**  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**  Character. The region of the AWS service.

**Value**

A list object or a character vector

**DhcpOptionsId**

The ID of the DHCP options set, or default to associate no DHCP options with the VPC.

**VpcId**

The ID of the VPC.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

**Description**

Associate Enclave Certificate Iam Role
Usage

```r
e2associate_enclave_certificate_iam_role(
  CertificateArn = NULL,
  RoleArn = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **CertificateArn**: Character. The ARN of the ACM certificate with which to associate the IAM role. [optional]
- **RoleArn**: Character. The ARN of the IAM role to associate with the ACM certificate. [optional]
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

- A list object or a character vector

**CertificateArn**

The ARN of the ACM certificate with which to associate the IAM role.

**RoleArn**

The ARN of the IAM role to associate with the ACM certificate. You can associate up to 16 IAM roles with an ACM certificate.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_associate_iam_instance_profile

Associate Iam Instance Profile

Description

Associates an IAM instance profile with a running or stopped instance. You cannot associate more than one IAM instance profile with an instance.

Usage

ec2_associate_iam_instance_profile(
    IamInstanceProfile,
    InstanceId,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

IamInstanceProfile
    Object. The IAM instance profile.
InstanceId
    Character. The ID of the instance.
simplify
    Logical. Whether to simplify the result and handle nextToken in the response[optional]
others
    Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error
    Logical. Whether to show an error message when a network error occurs.
retry_time
    Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout
    Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region
    Character. The region of the AWS service.
Value
A list object or a character vector

IamInstanceProfile
The IAM instance profile.

InstanceId
The ID of the instance.

Usage
ec2_associate_route_table(
  RouteTableId,
  DryRun = NULL,
  SubnetId = NULL,
  GatewayId = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments
  RouteTableId   Character. The ID of the route table.
  DryRun         Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
  SubnetId       Character. The ID of the subnet.[optional]
  GatewayId      Character. The ID of the internet gateway or virtual private gateway.[optional]
  simplify       Logical. Whether to simplify the result and handle nextToken in the response[optional]
  others         Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
  print_on_error Logical. Whether to show an error message when a network error occurs.
 Associate Subnet Cidr Block

Associates a CIDR block with your subnet. You can only associate a single IPv6 CIDR block with your subnet. An IPv6 CIDR block must have a prefix length of /64.

Usage

```r
ec2_associate_subnet_cidr_block(
  SubnetId,
  Ipv6CidrBlock,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
)```

**Value**

A list object or a character vector

**RouteTableId**

The ID of the route table.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

**SubnetId**

The ID of the subnet.

**GatewayId**

The ID of the internet gateway or virtual private gateway.
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

SubnetId Character. The ID of your subnet.
Ipv6CidrBlock Character. The IPv6 CIDR block for your subnet. The subnet must have a /64 prefix length.
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value

A list object or a character vector

SubnetId

The ID of your subnet.

Ipv6CidrBlock

The IPv6 CIDR block for your subnet. The subnet must have a /64 prefix length.

Description

Associate Transit Gateway Multicast Domain
Usage

ec2_associate_transit_gateway_multicast_domain(
    TransitGatewayMulticastDomainId = NULL,
    TransitGatewayAttachmentId = NULL,
    SubnetIds = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

TransitGatewayMulticastDomainId
    Character. The ID of the transit gateway multicast domain.[optional]

TransitGatewayAttachmentId
    Character. The ID of the transit gateway attachment to associate with the transit
    gateway multicast domain.[optional]

SubnetIds
    List. The IDs of the subnets to associate with the transit gateway multicast
    domain.[optional]

DryRun
    Logical. Checks whether you have the required permissions for the action, without
    actually making the request,...[optional]

simplify
    Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
    Named list. The parameters that are not included in the function parameters and
    need to be added into the request[optional]

print_on_error
    Logical. Whether to show an error message when a network error occurs.

retry_time
    Integer. Number of retries for a REST request when encounter the network
    issue. If the request has been sent retry_time times but still not be able to get
    the response, an error will be thrown.

network_timeout
    Numeric. Number of seconds to wait for a REST response until giving up. Can
    not be less than 1 ms.

region
    Character. The region of the AWS service.

Value

A list object or a character vector

TransitGatewayMulticastDomainId

The ID of the transit gateway multicast domain.
**TransitGatewayAttachmentId**

The ID of the transit gateway attachment to associate with the transit gateway multicast domain.

**SubnetIds**

The IDs of the subnets to associate with the transit gateway multicast domain.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**Description**

Associates the specified attachment with the specified transit gateway route table. You can associate only one route table with an attachment.

**Usage**

```r
ec2_associate_transit_gateway_route_table(
    TransitGatewayRouteTableId, 
    TransitGatewayAttachmentId, 
    DryRun = NULL, 
    simplify = TRUE, 
    others = list(), 
    print_on_error = aws_get_print_on_error(), 
    retry_time = aws_get_retry_time(), 
    network_timeout = aws_get_network_timeout(), 
    region = aws_get_region() 
)
```

**Arguments**

- **TransitGatewayRouteTableId**
  Character. The ID of the transit gateway route table.

- **TransitGatewayAttachmentId**
  Character. The ID of the attachment.

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]

- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response[optional]
ec2_associate_vpc_cidr_block

others

Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]

print_on_error

Logical. Whether to show an error message when a network error occurs.

retry_time

Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout

Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region

Character. The region of the AWS service.

Value

A list object or a character vector

TransitGatewayRouteTableId

The ID of the transit gateway route table.

TransitGatewayAttachmentId

The ID of the attachment.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_associate_vpc_cidr_block

Associate Vpc Cidr Block

Description

Associate Vpc Cidr Block

Usage

ec2_associate_vpc_cidr_block(
  VpcId,
  AmazonProvidedIpv6CidrBlock = NULL,
  CidrBlock = NULL,
  Ipv6CidrBlockNetworkBorderGroup = NULL,
  Ipv6Pool = NULL,
  Ipv6CidrBlock = NULL,
  simplify = TRUE,
ec2_associate_vpc_cidr_block

others = list(),
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

VpcId      Character. The ID of the VPC.
AmazonProvidedIpv6CidrBlock
            Logical. Requests an Amazon-provided IPv6 CIDR block with a /56 prefix length for the VPC.[optional]
CidrBlock  Character. An IPv4 CIDR block to associate with the VPC.[optional]
Ipv6CidrBlockNetworkBorderGroup
            Character. The name of the location from which we advertise the IPV6 CIDR block.[optional]
Ipv6Pool   Character. The ID of an IPv6 address pool from which to allocate the IPv6 CIDR block.[optional]
Ipv6CidrBlock
            Character. An IPv6 CIDR block from the IPv6 address pool.[optional]
simplify   Logical. Whether to simplify the result and handle nextToken in the response[optional]
others     Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error  Logical. Whether to show an error message when a network error occurs.
retry_time   Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region       Character. The region of the AWS service.

Value

A list object or a character vector

VpcId

The ID of the VPC.

AmazonProvidedIpv6CidrBlock

Requests an Amazon-provided IPv6 CIDR block with a /56 prefix length for the VPC. You cannot specify the range of IPv6 addresses, or the size of the CIDR block.
CidrBlock
An IPv4 CIDR block to associate with the VPC.

Ipv6CidrBlockNetworkBorderGroup
The name of the location from which we advertise the IPv6 CIDR block. Use this parameter to limit the CIDR block to this location.
You must set AmazonProvidedIpv6CidrBlock to true to use this parameter.
You can have one IPv6 CIDR block association per network border group.

Ipv6Pool
The ID of an IPv6 address pool from which to allocate the IPv6 CIDR block.

Ipv6CidrBlock
An IPv6 CIDR block from the IPv6 address pool. You must also specify Ipv6Pool in the request.
To let Amazon choose the IPv6 CIDR block for you, omit this parameter.

ec2_attach_classic_link_vpc
Attach Classic Link Vpc

Description
Attach Classic Link Vpc

Usage
ec2_attach_classic_link_vpc(
    SecurityGroupId,
    InstanceId,
    VpcId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()  
)
Arguments

SecurityGroupId
List. The ID of one or more of the VPC's security groups.

InstanceId
Character. The ID of an EC2-Classic instance to link to the ClassicLink-enabled VPC.

VpcId
Character. The ID of a ClassicLink-enabled VPC.

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response [optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value
A list object or a character vector

SecurityGroupId
The ID of one or more of the VPC's security groups. You cannot specify security groups from a different VPC.

InstanceId
The ID of an EC2-Classic instance to link to the ClassicLink-enabled VPC.

VpcId
The ID of a ClassicLink-enabled VPC.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
**ec2_attach_internet_gateway**

*Attach Internet Gateway*

**Description**

Attaches an internet gateway or a virtual private gateway to a VPC, enabling connectivity between the internet and the VPC. For more information about your VPC and internet gateway, see the Amazon Virtual Private Cloud User Guide.

**Usage**

```r
ec2_attach_internet_gateway(
  InternetGatewayId,
  VpcId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- **InternetGatewayId**
  Character. The ID of the internet gateway.

- **VpcId**
  Character. The ID of the VPC.

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]

- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response [optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encountering the network issue. If the request has been sent `retry_time` times but still not able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.

- **region**
  Character. The region of the AWS service.

**Value**

A list object or a character vector
InternetGatewayId

The ID of the internet gateway.

VpcId

The ID of the VPC.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Usage

ec2_attach_network_interface(
    DeviceIndex,
    InstanceId,
    NetworkInterfaceId,
    DryRun = NULL,
    NetworkCardIndex = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

DeviceIndex  Integer. The index of the device for the network interface attachment.
InstanceId   Character. The ID of the instance.
NetworkInterfaceId  Character. The ID of the network interface.
DryRun        Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]
NetworkCardIndex

Integer. The index of the network card. [optional]

simplify

Logical. Whether to simplify the result and handle nextToken in the response. [optional]

others

Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]

print_on_error

Logical. Whether to show an error message when a network error occurs.

retry_time

Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout

Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region

Character. The region of the AWS service.

Value

A list object or a character vector

DeviceIndex

The index of the device for the network interface attachment.

InstanceId

The ID of the instance.

NetworkInterfaceId

The ID of the network interface.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

NetworkCardIndex

The index of the network card. Some instance types support multiple network cards. The primary network interface must be assigned to network card index 0. The default is network card index 0.
ec2_attach_volume

Description

Attach Volume

Usage

ec2_attach_volume(
    Device,
    InstanceId,
    VolumeId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

Device Character. The device name (for example, /dev/sdh or xvdh).
InstanceId Character. The ID of the instance.
VolumeId Character. The ID of the EBS volume. The volume and instance must be within
    the same Availability Zone.
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network
    issue. If the request has been sent retry_time times but still not be able to get
    the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can
    not be less than 1 ms.
region Character. The region of the AWS service.

Value

A list object or a character vector
Device

The device name (for example, /dev/sdh or xvdh).

InstanceId

The ID of the instance.

VolumeId

The ID of the EBS volume. The volume and instance must be within the same Availability Zone.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**ec2_attach_vpn_gateway**

*Attach Vpn Gateway*

---

**Description**

Attach Vpn Gateway

**Usage**

```r
ec2_attach_vpn_gateway(
  VpcId,
  VpnGatewayId, 
  DryRun = NULL, 
  simplify = TRUE, 
  others = list(), 
  print_on_error = aws_get_print_on_error(), 
  retry_time = aws_get_retry_time(), 
  network_timeout = aws_get_network_timeout(), 
  region = aws_get_region() 
)
```

**Arguments**

- **VpcId**  Character. The ID of the VPC.
- **VpnGatewayId**  Character. The ID of the virtual private gateway.
- **DryRun**  Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]
- **simplify**  Logical. Whether to simplify the result and handle nextToken in the response[optional]
ec2_authorize_client_vpn_ingress

Authorize Client Vpn Ingress

Description

Adds an ingress authorization rule to a Client VPN endpoint. Ingress authorization rules act as firewall rules that grant access to networks. You must configure ingress authorization rules to enable clients to access resources in AWS or on-premises networks.

Usage

```r
ec2_authorize_client_vpn_ingress(
  ClientVpnEndpointId,
  TargetNetworkCidr,
  AccessGroupId = NULL,
  AuthorizeAllGroups = NULL,
  Description = NULL,
)```
ClientToken = NULL,
DryRun = NULL,
simplify = TRUE,
others = list(),
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

ClientVpnEndpointId
Character. The ID of the Client VPN endpoint.

TargetNetworkCidr
Character. The IPv4 address range, in CIDR notation, of the network for which
access is being authorized.

AccessGroupId
Character. The ID of the group to grant access to, for example, the Active Di-
rectory group or identity provider...[optional]

AuthorizeAllGroups
Logical. Indicates whether to grant access to all clients.[optional]

Description
Character. A brief description of the authorization rule.[optional]

ClientToken
Character. Unique, case-sensitive identifier that you provide to ensure the idem-
potency of the request.[optional]

DryRun
Logical. Checks whether you have the required permissions for the action, with-
out actually making the request,...[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and
need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network
issue. If the request has been sent retry_time times but still not be able to get
the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can
not be less than 1 ms.

region
Character. The region of the AWS service.

Value
A list object or a character vector

ClientVpnEndpointId
The ID of the Client VPN endpoint.
**TargetNetworkCidr**

The IPv4 address range, in CIDR notation, of the network for which access is being authorized.

**AccessGroupId**

The ID of the group to grant access to, for example, the Active Directory group or identity provider (IdP) group. Required if AuthorizeAllGroups is false or not specified.

**AuthorizeAllGroups**

Indicates whether to grant access to all clients. Specify true to grant all clients who successfully establish a VPN connection access to the network. Must be set to true if AccessGroupId is not specified.

**Description**

A brief description of the authorization rule.

**ClientToken**

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. For more information, see How to Ensure Idempotency.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

```python
ec2_authorize_security_group_egress

**Authorize Security Group Egress**

**Description**

Authorize Security Group Egress

**Usage**

```python
ec2_authorize_security_group_egress(
    GroupId,
    DryRun = NULL,
    IpPermissions = NULL,
    CidrIp = NULL,
    FromPort = NULL,
    IpProtocol = NULL,
    ToPort = NULL,
)```
SourceSecurityGroupName = NULL,
SourceSecurityGroupOwnerId = NULL,
simplify = TRUE,
others = list(),
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

GroupId Character. The ID of the security group.
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request,[optional]
IpPermissions List. The sets of IP permissions.[optional]
CidrIp Character. Not supported. Use a set of IP permissions to specify the CIDR.[optional]
FromPort Integer. Not supported. Use a set of IP permissions to specify the port.[optional]
IpProtocol Character. Not supported. Use a set of IP permissions to specify the protocol name or number.[optional]
ToPort Integer. Not supported. Use a set of IP permissions to specify the port.[optional]
SourceSecurityGroupName Character. Not supported. Use a set of IP permissions to specify a destination security group.[optional]
SourceSecurityGroupOwnerId Character. Not supported. Use a set of IP permissions to specify a destination security group.[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value

A list object or a character vector

GroupId

The ID of the security group.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

IpPermissions

The sets of IP permissions. You can't specify a destination security group and a CIDR IP address range in the same set of permissions.

CidrIp

Not supported. Use a set of IP permissions to specify the CIDR.

FromPort

Not supported. Use a set of IP permissions to specify the port.

IpProtocol

Not supported. Use a set of IP permissions to specify the protocol name or number.

ToPort

Not supported. Use a set of IP permissions to specify the port.

SourceSecurityGroupName

Not supported. Use a set of IP permissions to specify a destination security group.

SourceSecurityGroupOwnerId

Not supported. Use a set of IP permissions to specify a destination security group.
Usage

```r
ec2_authorize_security_group_ingress(
  CidrIp = NULL,
  FromPort = NULL,
  GroupId = NULL,
  GroupName = NULL,
  IpPermissions = NULL,
  IpProtocol = NULL,
  SourceSecurityGroupName = NULL,
  SourceSecurityGroupId = NULL,
  ToPort = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **CidrIp**: Character. The IPv4 address range, in CIDR format.[optional]
- **FromPort**: Integer. The start of port range for the TCP and UDP protocols, or an ICMP type number.[optional]
- **GroupId**: Character. The ID of the security group.[optional]
- **GroupName**: Character. [EC2-Classic, default VPC] The name of the security group.[optional]
- **IpPermissions**: List. The sets of IP permissions.[optional]
- **IpProtocol**: Character. The IP protocol name (tcp, udp, icmp) or number (see Protocol Numbers).[optional]
- **SourceSecurityGroupName**: Character. [EC2-Classic, default VPC] The name of the source security group.[optional]
- **SourceSecurityGroupId**: Character. [nondefault VPC] The AWS account ID for the source security group, if the source security group.[optional]
- **ToPort**: Integer. The end of port range for the TCP and UDP protocols, or an ICMP code number.[optional]
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
- **simplify**: Logical. Whether to simplify the result and handle nextToken in the response.[optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request.[optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  Character. The region of the AWS service.

Value

A list object or a character vector

CidrIp

The IPv4 address range, in CIDR format. You can’t specify this parameter when specifying a source security group. To specify an IPv6 address range, use a set of IP permissions.

Alternatively, use a set of IP permissions to specify multiple rules and a description for the rule.

FromPort

The start of port range for the TCP and UDP protocols, or an ICMP type number. For the ICMP type number, use -1 to specify all types. If you specify all ICMP types, you must specify all codes.

Alternatively, use a set of IP permissions to specify multiple rules and a description for the rule.

GroupId

The ID of the security group. You must specify either the security group ID or the security group name in the request. For security groups in a nondefault VPC, you must specify the security group ID.

GroupName

[EC2-Classic, default VPC] The name of the security group. You must specify either the security group ID or the security group name in the request.

IpPermissions

The sets of IP permissions.

IpProtocol

The IP protocol name (tcp, udp, icmp) or number (see Protocol Numbers). To specify icmpv6, use a set of IP permissions.

[VPC only] Use -1 to specify all protocols. If you specify -1 or a protocol other than tcp, udp, or icmp, traffic on all ports is allowed, regardless of any ports you specify.

Alternatively, use a set of IP permissions to specify multiple rules and a description for the rule.
SourceSecurityGroupName

[EC2-Classic, default VPC] The name of the source security group. You can't specify this parameter in combination with the following parameters: the CIDR IP address range, the start of the port range, the IP protocol, and the end of the port range. Creates rules that grant full ICMP, UDP, and TCP access. To create a rule with a specific IP protocol and port range, use a set of IP permissions instead. For EC2-VPC, the source security group must be in the same VPC.

SourceSecurityGroupOwnerId

[nondefault VPC] The AWS account ID for the source security group, if the source security group is in a different account. You can't specify this parameter in combination with the following parameters: the CIDR IP address range, the IP protocol, the start of the port range, and the end of the port range. Creates rules that grant full ICMP, UDP, and TCP access. To create a rule with a specific IP protocol and port range, use a set of IP permissions instead.

ToPort

The end of port range for the TCP and UDP protocols, or an ICMP code number. For the ICMP code number, use -1 to specify all codes. If you specify all ICMP types, you must specify all codes. Alternatively, use a set of IP permissions to specify multiple rules and a description for the rule.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**ec2_bundle_instance**  
*Bundle Instance*

**Description**

Bundle Instance

**Usage**

```r
ec2_bundle_instance(
    InstanceId,
    Storage,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```
Arguments

InstanceId    Character. The ID of the instance to bundle. Type: String Default: None Required: Yes
Storage       Object. The bucket in which to store the AMI.
DryRun        Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
simplify      Logical. Whether to simplify the result and handle nextToken in the response[optional]
others        Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time    Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region        Character. The region of the AWS service.

Value

A list object or a character vector

InstanceId

The ID of the instance to bundle.
Type: String
Default: None
Required: Yes

Storage

The bucket in which to store the AMI. You can specify a bucket that you already own or a new bucket that Amazon EC2 creates on your behalf. If you specify a bucket that belongs to someone else, Amazon EC2 returns an error.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
ec2_cancel_bundle_task

Cancel Bundle Task

Description

Cancels a bundling operation for an instance store-backed Windows instance.

Usage

```
ec2_cancel_bundle_task(
    BundleId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **BundleId**: Character. The ID of the bundle task.
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encountering the network issue. If the request has been sent `retry_time` times but still not able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

**BundleId**

The ID of the bundle task.
DryRun checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description
Cancel Capacity Reservation

Usage

ec2_cancel_capacity_reservation(
    CapacityReservationId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

CapacityReservationId
Character. The ID of the Capacity Reservation to be cancelled.

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.
Value

A list object or a character vector

CapacityReservationId

The ID of the Capacity Reservation to be cancelled.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

ec2_cancel_conversion_task

*Cancel Conversion Task*

Description

Cancel Conversion Task

Usage

```r
ec2_cancel_conversion_task(
    ConversionTaskId,
    DryRun = NULL,
    ReasonMessage = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConversionTaskId</td>
<td>Character. The ID of the conversion task.</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]</td>
</tr>
<tr>
<td>ReasonMessage</td>
<td>Character. The reason for canceling the conversion task.[optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response[optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
</tbody>
</table>
retry_time  Integer. Number of retries for a REST request when encountering the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.

region  Character. The region of the AWS service.

Value
A list object or a character vector

ConversionTaskId
The ID of the conversion task.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ReasonMessage
The reason for canceling the conversion task.

Description
Cancels an active export task. The request removes all artifacts of the export, including any partially-created Amazon S3 objects. If the export task is complete or is in the process of transferring the final disk image, the command fails and returns an error.

Usage
```r
ec2_cancel_export_task(  
  ExportTaskId,  
  simplify = TRUE,  
  others = list(),  
  print_on_error = aws_get_print_on_error(),  
  retry_time = aws_get_retry_time(),  
  network_timeout = aws_get_network_timeout(),  
  region = aws_get_region()  
)
```
Arguments

ExportTaskId     Character. The ID of the export task. This is the ID returned by CreateInstanceExportTask.
simplify        Logical. Whether to simplify the result and handle nextToken in the response[optional]
others           Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error   Logical. Whether to show an error message when a network error occurs.
retry_time       Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region           Character. The region of the AWS service.

Value

A list object or a character vector

ExportTaskId

The ID of the export task. This is the ID returned by CreateInstanceExportTask.

---

**ec2_cancel_import_task**

*Cancel Import Task*

---

Description

Cancels an in-process import virtual machine or import snapshot task.

Usage

```r
ec2_cancel_import_task(
    CancelReason = NULL,
    DryRun = NULL,
    ImportTaskId = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)```
Arguments

CancelReason  Character. The reason for canceling the task.[optional]
DryRun        Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
ImportTaskId  Character. The ID of the import image or import snapshot task to be canceled.[optional]
simplify      Logical. Whether to simplify the result and handle nextToken in the response[optional]
others        Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time    Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region        Character. The region of the AWS service.

Value

A list object or a character vector

CancelReason

The reason for canceling the task.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ImportTaskId

The ID of the import image or import snapshot task to be canceled.

Description

Cancel Reserved Instances Listing
Usage

```r
e2_cancel_reserved_instances_listing(
  ReservedInstancesListingId,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **ReservedInstancesListingId**
  Character. The ID of the Reserved Instance listing.

- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response (optional).

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request (optional).

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  Character. The region of the AWS service.

Value

A list object or a character vector

**ReservedInstancesListingId**

The ID of the Reserved Instance listing.

---

**ec2_cancel_spot_fleet_requests**

*Cancel Spot Fleet Requests*

Description

Cancel Spot Fleet Requests
ec2_cancel_spot_fleet_requests

Usage

ec2_cancel_spot_fleet_requests(  
  SpotFleetRequestId,  
  TerminateInstances,  
  DryRun = NULL,  
  simplify = TRUE,  
  others = list(),  
  print_on_error = aws_get_print_on_error(),  
  retry_time = aws_get_retry_time(),  
  network_timeout = aws_get_network_timeout(),  
  region = aws_get_region()  
)

Arguments

SpotFleetRequestId  
List. The IDs of the Spot Fleet requests.

TerminateInstances  
Logical. Indicates whether to terminate instances for a Spot Fleet request if it is canceled successfully.

DryRun  
Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]

simplify  
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others  
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error  
Logical. Whether to show an error message when a network error occurs.

retry_time  
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  
Character. The region of the AWS service.

Value

A list object or a character vector

SpotFleetRequestId

The IDs of the Spot Fleet requests.

TerminateInstances

Indicates whether to terminate instances for a Spot Fleet request if it is canceled successfully.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Usage

```r
dc2_cancel_spot_instance_requests(SpotInstanceRequestId, 
  DryRun = NULL, 
  simplify = TRUE, 
  others = list(), 
  print_on_error = aws_get_print_on_error(), 
  retry_time = aws_get_retry_time(), 
  network_timeout = aws_get_network_timeout(), 
  region = aws_get_region())
```

Arguments

- **SpotInstanceRequestId**
  - List. One or more Spot Instance request IDs.
- **DryRun**
  - Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
- **simplify**
  - Logical. Whether to simplify the result and handle `nextToken` in the response.[optional]
- **others**
  - Named list. The parameters that are not included in the function parameters and need to be added into the request.[optional]
- **print_on_error**
  - Logical. Whether to show an error message when a network error occurs.
- **retry_time**
  - Integer. Number of retries for a REST request when encountering the network issue. If the request has been sent `retry_time` times but is still not able to get the response, an error will be thrown.
- **network_timeout**
  - Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.
- **region**
  - Character. The region of the AWS service.
**ec2_confirm_product_instance**

**Value**
A list object or a character vector

**SpotInstanceRequestId**
One or more Spot Instance request IDs.

**DryRun**
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

**Description**
Determines whether a product code is associated with an instance. This action can only be used by the owner of the product code. It is useful when a product code owner must verify whether another user’s instance is eligible for support.

**Usage**

```r
ec2_confirm_product_instance(
  InstanceId,  
  ProductCode,  
  DryRun = NULL,  
  simplify = TRUE,  
  others = list(),  
  print_on_error = aws_get_print_on_error(),  
  retry_time = aws_get_retry_time(),  
  network_timeout = aws_get_network_timeout(),  
  region = aws_get_region()
)
```

**Arguments**

- **InstanceId** Character. The ID of the instance.
- **ProductCode** Character. The product code. This must be a product code that you own.
- **DryRun** Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]
- **simplify** Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others** Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
ec2_copy_fpga_image

print_on_error  Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region  Character. The region of the AWS service.

Value

A list object or a character vector

InstanceId

The ID of the instance.

ProductCode

The product code. This must be a product code that you own.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_copy_fpga_image  Copy Fpga Image

Description

Copies the specified Amazon FPGA Image (AFI) to the current Region.

Usage

```r
ec2_copy_fpga_image(
  SourceFpgaImageId,
  SourceRegion,
  DryRun = NULL,
  Description = NULL,
  Name = NULL,
  ClientToken = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
```
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

SourceFpgaImageId
Character. The ID of the source AFI.

SourceRegion
Character. The Region that contains the source AFI.

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]

Description
Character. The description for the new AFI.[optional]

Name
Character. The name for the new AFI. The default is the name of the source AFI.[optional]

ClientToken
Character. Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encountering the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value
A list object or a character vector

SourceFpgaImageId
The ID of the source AFI.

SourceRegion
The Region that contains the source AFI.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
Description

The description for the new AFI.

Name

The name for the new AFI. The default is the name of the source AFI.

ClientToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. For more information, see Ensuring Idempotency.

ec2_copy_image

Copy Image

Arguments

Name: Character. The name of the new AMI in the destination Region.
SourceImageId: Character. The ID of the AMI to copy.
SourceRegion: Character. The name of the Region that contains the AMI to copy.
ClientToken: Character. Unique, case-sensitive identifier you provide to ensure idempotency of the request.[optional]
Description
   Character. A description for the new AMI in the destination Region.[optional]

Encrypted
   Logical. Specifies whether the destination snapshots of the copied image should be encrypted.[optional]

KmsKeyId
   Character. The identifier of the symmetric AWS Key Management Service (AWS KMS) customer master key (CMK)...[optional]

DestinationOutpostArn
   Character. The Amazon Resource Name (ARN) of the Outpost to which to copy the AMI.[optional]

DryRun
   Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

simplify
   Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
   Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
   Logical. Whether to show an error message when a network error occurs.

retry_time
   Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
   Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
   Character. The region of the AWS service.

Value
   A list object or a character vector

Name
   The name of the new AMI in the destination Region.

SourceImageId
   The ID of the AMI to copy.

SourceRegion
   The name of the Region that contains the AMI to copy.

ClientToken
   Unique, case-sensitive identifier you provide to ensure idempotency of the request. For more information, see Ensuring idempotency in the Amazon EC2 API Reference.

Description
   A description for the new AMI in the destination Region.
Encrypted

Specifies whether the destination snapshots of the copied image should be encrypted. You can encrypt a copy of an unencrypted snapshot, but you cannot create an unencrypted copy of an encrypted snapshot. The default CMK for EBS is used unless you specify a non-default AWS Key Management Service (AWS KMS) CMK using KmsKeyId. For more information, see Amazon EBS Encryption in the Amazon Elastic Compute Cloud User Guide.

KmsKeyId

The identifier of the symmetric AWS Key Management Service (AWS KMS) customer master key (CMK) to use when creating encrypted volumes. If this parameter is not specified, your AWS managed CMK for EBS is used. If you specify a CMK, you must also set the encrypted state to true.

You can specify a CMK using any of the following:

- Key ID. For example, 1234abcd-12ab-34cd-56ef-1234567890ab.
- Key alias. For example, alias/ExampleAlias.
- Key ARN. For example, arn:aws:kms:us-east-1:012345678910:key/1234abcd-12ab-34cd-56ef-1234567890ab.
- Alias ARN. For example, arn:aws:kms:us-east-1:012345678910:alias/ExampleAlias.

AWS authenticates the CMK asynchronously. Therefore, if you specify an identifier that is not valid, the action can appear to complete, but eventually fails.

The specified CMK must exist in the destination Region.

Amazon EBS does not support asymmetric CMKs.

DestinationOutpostArn

The Amazon Resource Name (ARN) of the Outpost to which to copy the AMI. Only specify this parameter when copying an AMI from an AWS Region to an Outpost. The AMI must be in the Region of the destination Outpost. You cannot copy an AMI from an Outpost to a Region, from one Outpost to another, or within the same Outpost.

For more information, see Copying AMIs from an AWS Region to an Outpost in the Amazon Elastic Compute Cloud User Guide.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
**Description**

Copy Snapshot

**Usage**

```r
ec2_copy_snapshot(
  SourceRegion,
  SourceSnapshotId,
  Description = NULL,
  DestinationOutpostArn = NULL,
  DestinationRegion = NULL,
  Encrypted = NULL,
  KmsKeyId = NULL,
  PresignedUrl = NULL,
  TagSpecification = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- **SourceRegion**  
  Character. The ID of the Region that contains the snapshot to be copied.

- **SourceSnapshotId**  
  Character. The ID of the EBS snapshot to copy.

- **Description**  
  Character. A description for the EBS snapshot.[optional]

- **DestinationOutpostArn**  
  Character. The Amazon Resource Name (ARN) of the Outpost to which to copy the snapshot.[optional]

- **DestinationRegion**  
  Character. The destination Region to use in the PresignedUrl parameter of a snapshot copy operation.[optional]

- **Encrypted**  
  Logical. To encrypt a copy of an unencrypted snapshot if encryption by default is not enabled, enable encryption...[optional]

- **KmsKeyId**  
  Character. The identifier of the AWS Key Management Service (AWS KMS) customer master key (CMK) to use for...[optional]

- **PresignedUrl**  
  Character. When you copy an encrypted source snapshot using the Amazon EC2 Query API, you must supply a pre-signed...[optional]
TagSpecification
   List. The tags to apply to the new snapshot.[optional]

DryRun
   Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]

simplify
   Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
   Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
   Logical. Whether to show an error message when a network error occurs.

retry_time
   Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
   Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
   Character. The region of the AWS service.

Value
   A list object or a character vector

SourceRegion
   The ID of the Region that contains the snapshot to be copied.

SourceSnapshotId
   The ID of the EBS snapshot to copy.

Description
   A description for the EBS snapshot.

DestinationOutpostArn
   The Amazon Resource Name (ARN) of the Outpost to which to copy the snapshot. Only specify this parameter when copying a snapshot from an AWS Region to an Outpost. The snapshot must be in the Region for the destination Outpost. You cannot copy a snapshot from an Outpost to a Region, from one Outpost to another, or within the same Outpost.

   For more information, see Copying snapshots from an AWS Region to an Outpost in the Amazon Elastic Compute Cloud User Guide.

DestinationRegion
   The destination Region to use in the PresignedUrl parameter of a snapshot copy operation. This parameter is only valid for specifying the destination Region in a PresignedUrl parameter, where it is required.

   The snapshot copy is sent to the regional endpoint that you sent the HTTP request to (for example, ec2.us-east-1.amazonaws.com). With the AWS CLI, this is specified using the --region parameter or the default Region in your AWS configuration file.
Encrypted

To encrypt a copy of an unencrypted snapshot if encryption by default is not enabled, enable encryption using this parameter. Otherwise, omit this parameter. Encrypted snapshots are encrypted, even if you omit this parameter and encryption by default is not enabled. You cannot set this parameter to false. For more information, see Amazon EBS encryption in the Amazon Elastic Compute Cloud User Guide.

KmsKeyId

The identifier of the AWS Key Management Service (AWS KMS) customer master key (CMK) to use for Amazon EBS encryption. If this parameter is not specified, your AWS managed CMK for EBS is used. If KmsKeyId is specified, the encrypted state must be true.

You can specify the CMK using any of the following:

- Key ID. For example, 1234abcd-12ab-34cd-56ef-1234567890ab.
- Key alias. For example, alias/ExampleAlias.
- Key ARN. For example, arn:aws:kms:us-east-1:012345678910:key/1234abcd-12ab-34cd-56ef-1234567890ab.
- Alias ARN. For example, arn:aws:kms:us-east-1:012345678910:alias/ExampleAlias.

AWS authenticates the CMK asynchronously. Therefore, if you specify an ID, alias, or ARN that is not valid, the action can appear to complete, but eventually fails.

PresignedUrl

When you copy an encrypted source snapshot using the Amazon EC2 Query API, you must supply a pre-signed URL. This parameter is optional for unencrypted snapshots. For more information, see Query requests.

The PresignedUrl should use the snapshot source endpoint, the CopySnapshot action, and include the SourceRegion, SourceSnapshotId, and DestinationRegion parameters. The PresignedUrl must be signed using AWS Signature Version 4. Because EBS snapshots are stored in Amazon S3, the signing algorithm for this parameter uses the same logic that is described in Authenticating Requests: Using Query Parameters (AWS Signature Version 4) in the Amazon Simple Storage Service API Reference. An invalid or improperly signed PresignedUrl will cause the copy operation to fail asynchronously, and the snapshot will move to an error state.

TagSpecification

The tags to apply to the new snapshot.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
ec2_create_capacity_reservation

Create Capacity Reservation

Description
Create Capacity Reservation

Usage
ec2_create_capacity_reservation(
    InstanceType,
    InstancePlatform,
    InstanceCount,
    ClientToken = NULL,
    AvailabilityZone = NULL,
    AvailabilityZoneId = NULL,
    Tenancy = NULL,
    EbsOptimized = NULL,
    EphemeralStorage = NULL,
    EndDate = NULL,
    EndDateType = NULL,
    InstanceMatchCriteria = NULL,
    TagSpecifications = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>InstanceType</td>
<td>Character. The instance type for which to reserve capacity.</td>
</tr>
<tr>
<td>InstancePlatform</td>
<td>Character. The type of operating system for which to reserve capacity.</td>
</tr>
<tr>
<td>InstanceCount</td>
<td>Integer. The number of instances for which to reserve capacity.</td>
</tr>
<tr>
<td>ClientToken</td>
<td>Character. Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. [optional]</td>
</tr>
<tr>
<td>AvailabilityZone</td>
<td>Character. The Availability Zone in which to create the Capacity Reservation. [optional]</td>
</tr>
</tbody>
</table>
AvailabilityZoneId  
Character. The ID of the Availability Zone in which to create the Capacity Reservation.[optional]

Tenancy  
Character. Indicates the tenancy of the Capacity Reservation.[optional]

EbsOptimized  
Logical. Indicates whether the Capacity Reservation supports EBS-optimized instances.[optional]

EphemeralStorage  
Logical. Indicates whether the Capacity Reservation supports instances with temporary, block-level storage.[optional]

EndDate  
Character. The date and time at which the Capacity Reservation expires.[optional]

EndDateType  
Character. Indicates the way in which the Capacity Reservation ends.[optional]

InstanceMatchCriteria  
Character. Indicates the type of instance launches that the Capacity Reservation accepts.[optional]

TagSpecifications  
List. The tags to apply to the Capacity Reservation during launch.[optional]

DryRun  
Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]

simplify  
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others  
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error  
Logical. Whether to show an error message when a network error occurs.

retry_time  
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  
Character. The region of the AWS service.

Value  
A list object or a character vector

InstanceType  
The instance type for which to reserve capacity. For more information, see Instance types in the Amazon EC2 User Guide.

InstancePlatform  
The type of operating system for which to reserve capacity.

InstanceCount  
The number of instances for which to reserve capacity.
ClientToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. For more information, see Ensure Idempotency.

AvailabilityZone

The Availability Zone in which to create the Capacity Reservation.

AvailabilityZoneId

The ID of the Availability Zone in which to create the Capacity Reservation.

Tenancy

Indicates the tenancy of the Capacity Reservation. A Capacity Reservation can have one of the following tenancy settings:

- default - The Capacity Reservation is created on hardware that is shared with other AWS accounts.
- dedicated - The Capacity Reservation is created on single-tenant hardware that is dedicated to a single AWS account.

EbsOptimized

Indicates whether the Capacity Reservation supports EBS-optimized instances. This optimization provides dedicated throughput to Amazon EBS and an optimized configuration stack to provide optimal I/O performance. This optimization isn’t available with all instance types. Additional usage charges apply when using an EBS-optimized instance.

EphemeralStorage

Indicates whether the Capacity Reservation supports instances with temporary, block-level storage.

EndDate

The date and time at which the Capacity Reservation expires. When a Capacity Reservation expires, the reserved capacity is released and you can no longer launch instances into it. The Capacity Reservation’s state changes to expired when it reaches its end date and time.

You must provide an EndDate value if EndDateType is limited. Omit EndDate if EndDateType is unlimited.

If the EndDateType is limited, the Capacity Reservation is cancelled within an hour from the specified time. For example, if you specify 5/31/2019, 13:30:55, the Capacity Reservation is guaranteed to end between 13:30:55 and 14:30:55 on 5/31/2019.
**EndDateType**

Indicates the way in which the Capacity Reservation ends. A Capacity Reservation can have one of the following end types:

- **unlimited** - The Capacity Reservation remains active until you explicitly cancel it. Do not provide an EndDate if the EndDateType is unlimited.
- **limited** - The Capacity Reservation expires automatically at a specified date and time. You must provide an EndDate value if the EndDateType value is limited.

**InstanceMatchCriteria**

Indicates the type of instance launches that the Capacity Reservation accepts. The options include:

- **open** - The Capacity Reservation automatically matches all instances that have matching attributes (instance type, platform, and Availability Zone). Instances that have matching attributes run in the Capacity Reservation automatically without specifying any additional parameters.
- **targeted** - The Capacity Reservation only accepts instances that have matching attributes (instance type, platform, and Availability Zone), and explicitly target the Capacity Reservation. This ensures that only permitted instances can use the reserved capacity.

Default: open

**TagSpecifications**

The tags to apply to the Capacity Reservation during launch.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**ec2_create_carrier_gateway**

Create Carrier Gateway

**Description**

Creates a carrier gateway. For more information about carrier gateways, see Carrier gateways in the AWS Wavelength Developer Guide.
Usage

```r
ec2_create_carrier_gateway(
  VpcId,
  TagSpecification = NULL,
  DryRun = NULL,
  ClientToken = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **VpcId**: Character. The ID of the VPC to associate with the carrier gateway.
- **TagSpecification**: List. The tags to associate with the carrier gateway. [optional]
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **ClientToken**: Character. Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. [optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encountering a network issue. If the request has been sent `retry_time` times but still not able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

**VpcId**

The ID of the VPC to associate with the carrier gateway.

**TagSpecification**

The tags to associate with the carrier gateway.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ClientToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. For more information, see How to Ensure Idempotency.

ec2_create_client_vpn_endpoint

Create Client Vpn Endpoint

Description

Creates a Client VPN endpoint. A Client VPN endpoint is the resource you create and configure to enable and manage client VPN sessions. It is the destination endpoint at which all client VPN sessions are terminated.

Usage

ec2_create_client_vpn_endpoint(
    ClientCidrBlock,
    ServerCertificateArn,
    Authentication,
    ConnectionLogOptions,
    DnsServers = NULL,
    TransportProtocol = NULL,
    VpnPort = NULL,
    Description = NULL,
    SplitTunnel = NULL,
    DryRun = NULL,
    ClientToken = NULL,
    TagSpecification = NULL,
    SecurityGroupId = NULL,
    VpcId = NULL,
    SelfServicePortal = NULL,
    ClientConnectOptions = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
Arguments

ClientCidrBlock  
Character. The IPv4 address range, in CIDR notation, from which to assign client IP addresses.

ServerCertificateArn  
Character. The ARN of the server certificate.

Authentication  
List. Information about the authentication method to be used to authenticate clients.

ConnectionLogOptions  
Object. Information about the client connection logging options.

DnsServers  
List. Information about the DNS servers to be used for DNS resolution.[optional]

TransportProtocol  
Character. The transport protocol to be used by the VPN session. Default value: udp [optional]

VpnPort  
Integer. The port number to assign to the Client VPN endpoint for TCP and UDP traffic.[optional]

Description  
Character. A brief description of the Client VPN endpoint.[optional]

SplitTunnel  
Logical. Indicates whether split-tunnel is enabled on the AWS Client VPN endpoint.[optional]

DryRun  
Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

ClientToken  
Character. Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.[optional]

TagSpecification  
List. The tags to apply to the Client VPN endpoint during creation.[optional]

SecurityGroupId  
List. The IDs of one or more security groups to apply to the target network.[optional]

VpcId  
Character. The ID of the VPC to associate with the Client VPN endpoint.[optional]

SelfServicePortal  
Character. Specify whether to enable the self-service portal for the Client VPN endpoint.[optional]

ClientConnectOptions  
Object. The options for managing connection authorization for new client connections.[optional]

simplify  
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others  
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error  
Logical. Whether to show an error message when a network error occurs.

retry_time  
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not able to get the response, an error will be thrown.

network_timeout  
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  
Character. The region of the AWS service.
Value

A list object or a character vector

ClientCidrBlock

The IPv4 address range, in CIDR notation, from which to assign client IP addresses. The address range cannot overlap with the local CIDR of the VPC in which the associated subnet is located, or the routes that you add manually. The address range cannot be changed after the Client VPN endpoint has been created. The CIDR block should be /22 or greater.

ServerCertificateArn

The ARN of the server certificate. For more information, see the AWS Certificate Manager User Guide.

Authentication

Information about the authentication method to be used to authenticate clients.

ConnectionLogOptions

Information about the client connection logging options.

If you enable client connection logging, data about client connections is sent to a Cloudwatch Logs log stream. The following information is logged:

- Client connection requests
- Client connection results (successful and unsuccessful)
- Reasons for unsuccessful client connection requests
- Client connection termination time

DnsServers

Information about the DNS servers to be used for DNS resolution. A Client VPN endpoint can have up to two DNS servers. If no DNS server is specified, the DNS address configured on the device is used for the DNS server.

TransportProtocol

The transport protocol to be used by the VPN session.

Default value: udp

VpnPort

The port number to assign to the Client VPN endpoint for TCP and UDP traffic.

Valid Values: 443 | 1194

Default Value: 443
**Description**

A brief description of the Client VPN endpoint.

**SplitTunnel**

Indicates whether split-tunnel is enabled on the AWS Client VPN endpoint.

By default, split-tunnel on a VPN endpoint is disabled.

For information about split-tunnel VPN endpoints, see Split-Tunnel AWS Client VPN Endpoint in the AWS Client VPN Administrator Guide.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

**ClientToken**

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. For more information, see How to Ensure Idempotency.

**TagSpecification**

The tags to apply to the Client VPN endpoint during creation.

**SecurityGroupId**

The IDs of one or more security groups to apply to the target network. You must also specify the ID of the VPC that contains the security groups.

**VpcId**

The ID of the VPC to associate with the Client VPN endpoint. If no security group IDs are specified in the request, the default security group for the VPC is applied.

**SelfServicePortal**

Specify whether to enable the self-service portal for the Client VPN endpoint.

Default Value: enabled

**ClientConnectOptions**

The options for managing connection authorization for new client connections.
Description

Adds a route to a network to a Client VPN endpoint. Each Client VPN endpoint has a route table that describes the available destination network routes. Each route in the route table specifies the path for traffic to specific resources or networks.

Usage

```r
ec2_create_client_vpn_route(
  ClientVpnEndpointId,
  DestinationCidrBlock,
  TargetVpcSubnetId,
  Description = NULL,
  ClientToken = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **ClientVpnEndpointId**
  Character. The ID of the Client VPN endpoint to which to add the route.

- **DestinationCidrBlock**
  Character. The IPv4 address range, in CIDR notation, of the route destination.

- **TargetVpcSubnetId**
  Character. The ID of the subnet through which you want to route traffic.

- **Description**
  Character. A brief description of the route.[optional]

- **ClientToken**
  Character. Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.[optional]

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.
**retry_time**

Integer. Number of retries for a REST request when encountering the network issue. If the request has been sent `retry_time` times but still not able to get the response, an error will be thrown.

**network_timeout**

Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.

**region**

Character. The region of the AWS service.

**Value**

A list object or a character vector

**ClientVpnEndpointId**

The ID of the Client VPN endpoint to which to add the route.

**DestinationCidrBlock**

The IPv4 address range, in CIDR notation, of the route destination. For example:

- To add a route for Internet access, enter `0.0.0.0/0`
- To add a route for a peered VPC, enter the peered VPC’s IPv4 CIDR range
- To add a route for an on-premises network, enter the AWS Site-to-Site VPN connection’s IPv4 CIDR range
- To add a route for the local network, enter the client CIDR range

**TargetVpcSubnetId**

The ID of the subnet through which you want to route traffic. The specified subnet must be an existing target network of the Client VPN endpoint.

Alternatively, if you’re adding a route for the local network, specify `local`.

**Description**

A brief description of the route.

**ClientToken**

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. For more information, see How to Ensure Idempotency.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.  

---

**ec2_create_client_vpn_route**
ec2_create_customer_gateway

Create Customer Gateway

Description

Create Customer Gateway

Usage

ec2_create_customer_gateway(
    BgpAsn,
    Type,
    IpAddress = NULL,
    CertificateArn = NULL,
    TagSpecification = NULL,
    DeviceName = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

BgpAsn          Integer. For devices that support BGP, the customer gateway\'s BGP ASN. Default: 65000
Type            Character. The type of VPN connection that this customer gateway supports (ipsec.1).
IpAddress       Character. The Internet-routable IP address for the customer gateway\'s outside interface.[optional]
CertificateArn  Character. The Amazon Resource Name (ARN) for the customer gateway certificate.[optional]
TagSpecification List. The tags to apply to the customer gateway.[optional]
DeviceName      Character. A name for the customer gateway device. Length Constraints: Up to 255 characters. [optional]
DryRun          Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
simplify        Logical. Whether to simplify the result and handle nextToken in the response[optional]
others          Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
ec2_create_customer_gateway

print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value
A list object or a character vector

BgpAsn
For devices that support BGP, the customer gateway’s BGP ASN.
Default: 65000

Type
The type of VPN connection that this customer gateway supports (ipsec.1).

IpAddress
The Internet-routable IP address for the customer gateway’s outside interface. The address must be static.

CertificateArn
The Amazon Resource Name (ARN) for the customer gateway certificate.

TagSpecification
The tags to apply to the customer gateway.

DeviceName
A name for the customer gateway device.
Length Constraints: Up to 255 characters.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
ec2_create_default_subnet

Create Default Subnet

Description

Creates a default subnet with a size /20 IPv4 CIDR block in the specified Availability Zone in your default VPC. You can have only one default subnet per Availability Zone. For more information, see Creating a Default Subnet in the Amazon Virtual Private Cloud User Guide.

Usage

ec2_create_default_subnet(
    AvailabilityZone,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

AvailabilityZone  Character. The Availability Zone in which to create the default subnet.
DryRun  Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
simplify  Logical. Whether to simplify the result and handle nextToken in the response[optional]
others  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error  Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encountering the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.
region  Character. The region of the AWS service.

Value

A list object or a character vector
AvailabilityZone

The Availability Zone in which to create the default subnet.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

desc = "Create Default Vpc"

Usage

ec2_create_default_vpc(
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request..[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.
Value
A list object or a character vector

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_create_dhcp_options
Create Dhcp Options

Description
Create Dhcp Options

Usage
ec2_create_dhcp_options(
  DhcpConfiguration,
  TagSpecification = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments
DhcpConfiguration
  List. A DHCP configuration option.
TagSpecification
  List. The tags to assign to the DHCP option.[optional]
DryRun
  Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]
simplify
  Logical. Whether to simplify the result and handle nextToken in the response[optional]
others
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error
  Logical. Whether to show an error message when a network error occurs.
retry_time
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value
A list object or a character vector

DhcpConfiguration
A DHCP configuration option.

TagSpecification
The tags to assign to the DHCP option.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
Arguments

VpcId    Character. The ID of the VPC for which to create the egress-only internet gateway.

ClientToken    Character. Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.[optional]

DryRun   Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

TagSpecification

List. The tags to assign to the egress-only internet gateway.[optional]

simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]

others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.

Value

A list object or a character vector

VpcId

The ID of the VPC for which to create the egress-only internet gateway.

ClientToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. For more information, see How to Ensure Idempotency.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

TagSpecification

The tags to assign to the egress-only internet gateway.
ec2_create_fleet  Create Fleet

Description

Create Fleet

Usage

ec2_create_fleet(
    LaunchTemplateConfigs,  
    TargetCapacitySpecification, 
    DryRun = NULL, 
    ClientToken = NULL, 
    SpotOptions = NULL, 
    OnDemandOptions = NULL, 
    ExcessCapacityTerminationPolicy = NULL, 
    TerminateInstancesWithExpiration = NULL, 
    Type = NULL, 
    ValidFrom = NULL, 
    ValidUntil = NULL, 
    ReplaceUnhealthyInstances = NULL, 
    TagSpecification = NULL, 
    simplify = TRUE, 
    others = list(), 
    print_on_error = aws_get_print_on_error(), 
    retry_time = aws_get_retry_time(), 
    network_timeout = aws_get_network_timeout(), 
    region = aws_get_region()
)

Arguments

LaunchTemplateConfigs
    List. The configuration for the EC2 Fleet.

TargetCapacitySpecification
    Object. The number of units to request.

DryRun
    Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

ClientToken
    Character. Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.[optional]

SpotOptions
    Object. Describes the configuration of Spot Instances in an EC2 Fleet.[optional]

OnDemandOptions
    Object. Describes the configuration of On-Demand Instances in an EC2 Fleet.[optional]
ExcessCapacityTerminationPolicy
   Character. Indicates whether running instances should be terminated if the total target capacity of the EC2...[optional]

TerminateInstancesWithExpiration
   Logical. Indicates whether running instances should be terminated when the EC2 Fleet expires.[optional]

Type
   Character. The type of request.[optional]

ValidFrom
   Character. The start date and time of the request, in UTC format (for example, YYYY-MM-DDTHH:MM:SSZ)....[optional]

ValidUntil
   Character. The end date and time of the request, in UTC format (for example, YYYY-MM-DDTHH:MM:SSZ)....[optional]

ReplaceUnhealthyInstances
   Logical. Indicates whether EC2 Fleet should replace unhealthy Spot Instances.[optional]

TagSpecification
   List. The key-value pair for tagging the EC2 Fleet request on creation.[optional]

simplify
   Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
   Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
   Logical. Whether to show an error message when a network error occurs.

retry_time
   Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
   Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
   Character. The region of the AWS service.

Value
   A list object or a character vector

LaunchTemplateConfigs
   The configuration for the EC2 Fleet.

TargetCapacitySpecification
   The number of units to request.

DryRun
   Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
ClientToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. For more information, see Ensuring Idempotency.

SpotOptions

Describes the configuration of Spot Instances in an EC2 Fleet.

OnDemandOptions

Describes the configuration of On-Demand Instances in an EC2 Fleet.

ExcessCapacityTerminationPolicy

Indicates whether running instances should be terminated if the total target capacity of the EC2 Fleet is decreased below the current size of the EC2 Fleet.

TerminateInstancesWithExpiration

Indicates whether running instances should be terminated when the EC2 Fleet expires.

Type

The type of request. The default value is maintain.

- maintain - The EC2 Fleet places an asynchronous request for your desired capacity, and continues to maintain your desired Spot capacity by replenishing interrupted Spot Instances.
- request - The EC2 Fleet places an asynchronous one-time request for your desired capacity, but does submit Spot requests in alternative capacity pools if Spot capacity is unavailable, and does not maintain Spot capacity if Spot Instances are interrupted.
- instant - The EC2 Fleet places a synchronous one-time request for your desired capacity, and returns errors for any instances that could not be launched.

For more information, see EC2 Fleet request types in the Amazon EC2 User Guide.

ValidFrom

The start date and time of the request, in UTC format (for example, YYYY-MM-DDTHH:MM:SSZ). The default is to start fulfilling the request immediately.

ValidUntil

The end date and time of the request, in UTC format (for example, YYYY-MM-DDTHH:MM:SSZ). At this point, no new EC2 Fleet requests are placed or able to fulfill the request. If no value is specified, the request remains until you cancel it.

ReplaceUnhealthyInstances

Indicates whether EC2 Fleet should replace unhealthy Spot Instances. Supported only for fleets of type maintain. For more information, see EC2 Fleet health checks in the Amazon EC2 User Guide.
ec2_create_flow_logs

TagSpecification

The key-value pair for tagging the EC2 Fleet request on creation. The value for ResourceType must be fleet, otherwise the fleet request fails. To tag instances at launch, specify the tags in the launch template. For information about tagging after launch, see Tagging your resources.

ec2_create_flow_logs  Create Flow Logs

Description

Create Flow Logs

Usage

ec2_create_flow_logs(
  ResourceId,
  ResourceType,
  TrafficType,
  DryRun = NULL,
  ClientToken = NULL,
  DeliverLogsPermissionArn = NULL,
  LogGroupName = NULL,
  LogDestinationType = NULL,
  LogDestination = NULL,
  LogFormat = NULL,
  TagSpecification = NULL,
  MaxAggregationInterval = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

ResourceId  List. The ID of the subnet, network interface, or VPC for which you want to create a flow log.
ResourceId  Character. The type of resource for which to create the flow log.
TrafficType  Character. The type of traffic to log.
DryRun  Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
ClientToken  Character. Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.[optional]
ec2_create_flow_logs

DeliverLogsPermissionArn
   Character. The ARN for the IAM role that permits Amazon EC2 to publish flow
   logs to a CloudWatch Logs log...[optional]

LogGroupName
   Character. The name of a new or existing CloudWatch Logs log group where
   Amazon EC2 publishes your flow logs...[optional]

LogDestinationType
   Character. Specifies the type of destination to which the flow log data is to be
   published.[optional]

LogDestination
   Character. Specifies the destination to which the flow log data is to be pub-
   lished.[optional]

LogFormat
   Character. The fields to include in the flow log record, in the order in which they
   should appear.[optional]

TagSpecification
   List. The tags to apply to the flow logs.[optional]

MaxAggregationInterval
   Integer. The maximum interval of time during which a flow of packets is cap-
   tured and aggregated into a flow...[optional]

simplify
   Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
   Named list. The parameters that are not included in the function parameters and
   need to be added into the request[optional]

print_on_error
   Logical. Whether to show an error message when a network error occurs.

retry_time
   Integer. Number of retries for a REST request when encounter the network
   issue. If the request has been sent retry_time times but still not be able to get
   the response, an error will be thrown.

network_timeout
   Numeric. Number of seconds to wait for a REST response until giving up. Can
   not be less than 1 ms.

region
   Character. The region of the AWS service.

Value
   A list object or a character vector

ResourceId
   The ID of the subnet, network interface, or VPC for which you want to create a flow log.
   Constraints: Maximum of 1000 resources

ResourceType
   The type of resource for which to create the flow log. For example, if you specified a VPC ID for
   the ResourceId property, specify VPC for this property.

TrafficType
   The type of traffic to log. You can log traffic that the resource accepts or rejects, or all traffic.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ClientToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. For more information, see How to Ensure Idempotency.

DeliverLogsPermissionArn

The ARN for the IAM role that permits Amazon EC2 to publish flow logs to a CloudWatch Logs log group in your account.

If you specify LogDestinationType as s3, do not specify DeliverLogsPermissionArn or LogGroupName.

LogGroupName

The name of a new or existing CloudWatch Logs log group where Amazon EC2 publishes your flow logs.

If you specify LogDestinationType as s3, do not specify DeliverLogsPermissionArn or LogGroupName.

LogDestinationType

Specifies the type of destination to which the flow log data is to be published. Flow log data can be published to CloudWatch Logs or Amazon S3. To publish flow log data to CloudWatch Logs, specify cloud-watch-logs. To publish flow log data to Amazon S3, specify s3.

If you specify LogDestinationType as s3, do not specify DeliverLogsPermissionArn or LogGroupName.

Default: cloud-watch-logs

LogDestination

Specifies the destination to which the flow log data is to be published. Flow log data can be published to a CloudWatch Logs log group or an Amazon S3 bucket. The value specified for this parameter depends on the value specified for LogDestinationType.

If LogDestinationType is not specified or cloud-watch-logs, specify the Amazon Resource Name (ARN) of the CloudWatch Logs log group. For example, to publish to a log group called my-logs, specify arn:aws:logs:us-east-1:123456789012:log-group:my-logs. Alternatively, use LogGroupName instead.

If LogDestinationType is s3, specify the ARN of the Amazon S3 bucket. You can also specify a subfolder in the bucket. To specify a subfolder in the bucket, use the following ARN format: bucket_ARN/subfolder_name/. For example, to specify a subfolder named my-logs in a bucket named my-bucket, use the following ARN: arn:aws:s3:::my-bucket/my-logs/. You cannot use AWSLogs as a subfolder name. This is a reserved term.
LogFormat

The fields to include in the flow log record, in the order in which they should appear. For a list of available fields, see Flow Log Records. If you omit this parameter, the flow log is created using the default format. If you specify this parameter, you must specify at least one field.

Specify the fields using the \$\{field-id\} format, separated by spaces. For the AWS CLI, use single quotation marks (\' \') to surround the parameter value.

TagSpecification

The tags to apply to the flow logs.

MaxAggregationInterval

The maximum interval of time during which a flow of packets is captured and aggregated into a flow log record. You can specify 60 seconds (1 minute) or 600 seconds (10 minutes).

When a network interface is attached to a Nitro-based instance, the aggregation interval is always 60 seconds or less, regardless of the value that you specify.

Default: 600

ec2_create_fpga_image  Create Fpga Image

Description

Create Fpga Image

Usage

ec2_create_fpga_image(
  InputStorageLocation,
  DryRun = NULL,
  LogsStorageLocation = NULL,
  Description = NULL,
  Name = NULL,
  ClientToken = NULL,
  TagSpecification = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
Arguments

**InputStorageLocation**
Object. The location of the encrypted design checkpoint in Amazon S3. The input must be a tarball.

**DryRun**
Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]

**LogsStorageLocation**
Object. The location in Amazon S3 for the output logs. [optional]

**Description**
Character. A description for the AFI. [optional]

**Name**
Character. A name for the AFI. [optional]

**ClientToken**
Character. Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. [optional]

**TagSpecification**
List. The tags to apply to the FPGA image during creation. [optional]

**simplify**
Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]

**others**
Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]

**print_on_error**
Logical. Whether to show an error message when a network error occurs.

**retry_time**
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

**network_timeout**
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

**region**
Character. The region of the AWS service.

Value

A list object or a character vector

**InputStorageLocation**

The location of the encrypted design checkpoint in Amazon S3. The input must be a tarball.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

**LogsStorageLocation**

The location in Amazon S3 for the output logs.

**Description**

A description for the AFI.
Name
A name for the AFI.

ClientToken
Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. For more information, see Ensuring Idempotency.

TagSpecification
The tags to apply to the FPGA image during creation.

```r
ec2_create_image

Arguments
InstanceId Character. The ID of the instance.
Name Character. A name for the new image.
BlockDeviceMapping List. The block device mappings.[optional]
Description Character. A description for the new image.[optional]
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
```
NoReboot  Logical. By default, Amazon EC2 attempts to shut down and reboot the instance before creating the image.[optional]

TagSpecification  List. The tags to apply to the AMI and snapshots on creation.[optional]

simplify  Logical. Whether to simplify the result and handle nextToken in the response[optional]

others  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error  Logical. Whether to show an error message when a network error occurs.

retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  Character. The region of the AWS service.

Value  A list object or a character vector

InstanceId  The ID of the instance.

Name  A name for the new image.

Constraints: 3-128 alphanumeric characters, parentheses (()), square brackets ([]), spaces ( ), periods (.), slashes (/), dashes (-), single quotes (‘), at-signs (@), or underscores(_)

BlockDeviceMapping  The block device mappings. This parameter cannot be used to modify the encryption status of existing volumes or snapshots. To create an AMI with encrypted snapshots, use the CopyImage action.

Description  A description for the new image.

DryRun  Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
NoReboot

By default, Amazon EC2 attempts to shut down and reboot the instance before creating the image. If the No Reboot option is set, Amazon EC2 doesn't shut down the instance before creating the image. When this option is used, file system integrity on the created image can't be guaranteed.

TagSpecification

The tags to apply to the AMI and snapshots on creation. You can tag the AMI, the snapshots, or both.

- To tag the AMI, the value for ResourceType must be image.
- To tag the snapshots that are created of the root volume and of other EBS volumes that are attached to the instance, the value for ResourceType must be snapshot. The same tag is applied to all of the snapshots that are created.

If you specify other values for ResourceType, the request fails.
To tag an AMI or snapshot after it has been created, see CreateTags.

```r
ec2_create_instance_export_task

Create Instance Export Task

Description

Create Instance Export Task

Usage

ec2_create_instance_export_task(
    ExportToS3, 
    InstanceId, 
    TargetEnvironment, 
    Description = NULL, 
    TagSpecification = NULL, 
    simplify = TRUE, 
    others = list(), 
    print_on_error = aws_get_print_on_error(), 
    retry_time = aws_get_retry_time(), 
    network_timeout = aws_get_network_timeout(), 
    region = aws_get_region()
)
```
Arguments

- **ExportToS3**: Object. The format and location for an export instance task.
- **InstanceId**: Character. The ID of the instance.
- **TargetEnvironment**: Character. The target virtualization environment.
- **Description**: Character. A description for the conversion task or the resource being exported. Optional.
- **TagSpecification**: List. The tags to apply to the export instance task during creation. Optional.
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response. Optional.
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request. Optional.
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encountering the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

**ExportToS3**

The format and location for an export instance task.

**InstanceId**

The ID of the instance.

**TargetEnvironment**

The target virtualization environment.

**Description**

A description for the conversion task or the resource being exported. The maximum length is 255 characters.

**TagSpecification**

The tags to apply to the export instance task during creation.
ec2_create_internet_gateway

Create Internet Gateway

Description

Create Internet Gateway

Usage

ec2_create_internet_gateway(
    TagSpecification = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

TagSpecification
List. The tags to assign to the internet gateway.[optional]

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value

A list object or a character vector

TagSpecification

The tags to assign to the internet gateway.
DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Usage

ec2_create_key_pair(
    KeyName,
    DryRun = NULL,
    TagSpecification = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

KeyName Character. A unique name for the key pair. Constraints: Up to 255 ASCII characters
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request,[optional]
TagSpecification List. The tags to apply to the new key pair.[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.
Value
A list object or a character vector

KeyName
A unique name for the key pair.
Constraints: Up to 255 ASCII characters

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

TagSpecification
The tags to apply to the new key pair.

---

**ec2_create_launch_template**

*Create Launch Template*

---

**Description**

Creates a launch template. A launch template contains the parameters to launch an instance. When you launch an instance using RunInstances, you can specify a launch template instead of providing the launch parameters in the request. For more information, see Launching an instance from a launch template in the Amazon Elastic Compute Cloud User Guide.

**Usage**

```r
ec2_create_launch_template(
  LaunchTemplateName,
  LaunchTemplateData,
  DryRun = NULL,
  ClientToken = NULL,
  VersionDescription = NULL,
  TagSpecification = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
Arguments

LaunchTemplateName
Character. A name for the launch template.

LaunchTemplateData
Object. The information for the launch template.

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

ClientToken
Character. Unique, case-sensitive identifier you provide to ensure the idempotency of the request.[optional]

VersionDescription
Character. A description for the first version of the launch template.[optional]

TagSpecification
List. The tags to apply to the launch template during creation.[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value

A list object or a character vector

LaunchTemplateName

A name for the launch template.

LaunchTemplateData

The information for the launch template.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
**ClientToken**

Unique, case-sensitive identifier you provide to ensure the idempotency of the request. For more information, see [Ensuring Idempotency](#).

Constraint: Maximum 128 ASCII characters.

**VersionDescription**

A description for the first version of the launch template.

**TagSpecification**

The tags to apply to the launch template during creation.

---

```r
ec2_create_launch_template_version

Create Launch Template Version

Description

Create Launch Template Version

Usage

e2_create_launch_template_version(
    LaunchTemplateData,
    DryRun = NULL,
    ClientToken = NULL,
    LaunchTemplateId = NULL,
    LaunchTemplateName = NULL,
    SourceVersion = NULL,
    VersionDescription = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

**Arguments**

- **LaunchTemplateData**
  - Object. The information for the launch template.

- **DryRun**
  - Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
ec2_create_launch_template_version

ClientToken  Character. Unique, case-sensitive identifier you provide to ensure the idempotency of the request.[optional]
LaunchTemplateId  Character. The ID of the launch template.[optional]
LaunchTemplateName  Character. The name of the launch template.[optional]
SourceVersion  Character. The version number of the launch template version on which to base the new version.[optional]
VersionDescription  Character. A description for the version of the launch template.[optional]
simplify  Logical. Whether to simplify the result and handle nextToken in the response[optional]
others  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error  Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region  Character. The region of the AWS service.

Value
A list object or a character vector

LaunchTemplateData
The information for the launch template.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ClientToken
Unique, case-sensitive identifier you provide to ensure the idempotency of the request. For more information, see Ensuring Idempotency.
Constraint: Maximum 128 ASCII characters.

LaunchTemplateId
The ID of the launch template. You must specify either the launch template ID or launch template name in the request.
LaunchTemplateName

The name of the launch template. You must specify either the launch template ID or launch template name in the request.

SourceVersion

The version number of the launch template version on which to base the new version. The new version inherits the same launch parameters as the source version, except for parameters that you specify in LaunchTemplateData. Snapshots applied to the block device mapping are ignored when creating a new version unless they are explicitly included.

VersionDescription

A description for the version of the launch template.

ec2_create_local_gateway_route

Create Local Gateway Route

Description

Creates a static route for the specified local gateway route table.

Usage

```r
ec2_create_local_gateway_route(
    DestinationCidrBlock,
    LocalGatewayRouteTableId,
    LocalGatewayVirtualInterfaceGroupId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **DestinationCidrBlock**
  - Character. The CIDR range used for destination matches.
- **LocalGatewayRouteTableId**
  - Character. The ID of the local gateway route table.
- **LocalGatewayVirtualInterfaceGroupId**
  - Character. The ID of the virtual interface group.
DryRun  Logical. Checks whether you have the required permissions for the action, without actually making the request, [optional]

simplify  Logical. Whether to simplify the result and handle nextToken in the response [optional]

others  Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]

print_on_error  Logical. Whether to show an error message when a network error occurs.

retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  Character. The region of the AWS service.

Value
A list object or a character vector

DestinationCidrBlock
The CIDR range used for destination matches. Routing decisions are based on the most specific match.

LocalGatewayRouteTableId
The ID of the local gateway route table.

LocalGatewayVirtualInterfaceGroupId
The ID of the virtual interface group.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_create_local_gateway_route_table_vpc_association

Create Local Gateway Route Table Vpc Association

Description
Associates the specified VPC with the specified local gateway route table.
Usage

```r
e2.create_local_gateway_route_table_vpc_association(
  LocalGatewayRouteTableId, VpcId,
  TagSpecification = NULL, DryRun = NULL,
  simplify = TRUE, others = list(),
  print_on_error = aws.get_print_on_error(),
  retry_time = aws.get_retry_time(),
  network_timeout = aws.get_network_timeout(),
  region = aws.get_region()
)
```

Arguments

- `LocalGatewayRouteTableId`  
  Character. The ID of the local gateway route table.
- `VpcId`  
  Character. The ID of the VPC.
- `TagSpecification`  
  List. The tags to assign to the local gateway route table VPC association.[optional]
- `DryRun`  
  Logical. Checks whether you have the required permissions for the action, without actually making the request,[optional]
- `simplify`  
  Logical. Whether to simplify the result and handle `nextToken` in the response[optional]
- `others`  
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- `print_on_error`  
  Logical. Whether to show an error message when a network error occurs.
- `retry_time`  
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- `network_timeout`  
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- `region`  
  Character. The region of the AWS service.

Value

A list object or a character vector

**LocalGatewayRouteTableId**  
The ID of the local gateway route table.

**VpcId**  
The ID of the VPC.
TagSpecification
The tags to assign to the local gateway route table VPC association.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_create_managed_prefix_list
Create Managed Prefix List

Description
Create Managed Prefix List

Usage
ec2_create_managed_prefix_list(
  PrefixListName,
  MaxEntries,
  AddressFamily,
  DryRun = NULL,
  Entry = NULL,
  TagSpecification = NULL,
  ClientToken = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments
PrefixListName  Character. A name for the prefix list.
MaxEntries      Integer. The maximum number of entries for the prefix list.
AddressFamily   Character. The IP address type. Valid Values: IPv4 | IPv6
DryRun          Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]
Entry           List. One or more entries for the prefix list.[optional]
TagSpecification List. The tags to apply to the prefix list during creation.[optional]
ec2_create_managed_prefix_list

ClientToken  Character. Unique, case-sensitive identifier you provide to ensure the idempotency of the request.[optional]
simplify  Logical. Whether to simplify the result and handle nextToken in the response[optional]
others  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error  Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region  Character. The region of the AWS service.

Value

A list object or a character vector

PrefixListName

A name for the prefix list.
Constraints: Up to 255 characters in length. The name cannot start with com.amazonaws.

MaxEntries

The maximum number of entries for the prefix list.

AddressFamily

The IP address type.
Valid Values: IPv4 \ IPv6

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Entry

One or more entries for the prefix list.

TagSpecification

The tags to apply to the prefix list during creation.
ClientToken

Unique, case-sensitive identifier you provide to ensure the idempotency of the request. For more information, see Ensuring Idempotency.

Constraints: Up to 255 UTF-8 characters in length.

ec2_create_nat_gateway

Create Nat Gateway

Description

Creates a NAT gateway in the specified public subnet. This action creates a network interface in the specified subnet with a private IP address from the IP address range of the subnet. Internet-bound traffic from a private subnet can be routed to the NAT gateway, therefore enabling instances in the private subnet to connect to the internet. For more information, see NAT Gateways in the Amazon Virtual Private Cloud User Guide.

Usage

ec2_create_nat_gateway(
  SubnetId,
  AllocationId,
  ClientToken = NULL,
  DryRun = NULL,
  TagSpecification = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SubnetId</td>
<td>Character. The subnet in which to create the NAT gateway.</td>
</tr>
<tr>
<td>AllocationId</td>
<td>Character. The allocation ID of an Elastic IP address to associate with the NAT gateway.</td>
</tr>
<tr>
<td>ClientToken</td>
<td>Character. Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.[optional]</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]</td>
</tr>
<tr>
<td>TagSpecification</td>
<td>List. The tags to assign to the NAT gateway.[optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response[optional]</td>
</tr>
</tbody>
</table>
others  
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error  
Logical. Whether to show an error message when a network error occurs.

retry_time  
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  
Character. The region of the AWS service.

Value

A list object or a character vector

SubnetId

The subnet in which to create the NAT gateway.

AllocationId

The allocation ID of an Elastic IP address to associate with the NAT gateway. If the Elastic IP address is associated with another resource, you must first disassociate it.

ClientToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. For more information, see How to Ensure Idempotency.

Constraint: Maximum 64 ASCII characters.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

TagSpecification

The tags to assign to the NAT gateway.
**ec2_create_network_acl**

*Create Network Acl*

**Description**
Create Network Acl

**Usage**
```r
e2_create_network_acl(  VpcId,  DryRun = NULL,  TagSpecification = NULL,  simplify = TRUE,  others = list(),  print_on_error = aws_get_print_on_error(),  retry_time = aws_get_retry_time(),  network_timeout = aws_get_network_timeout(),  region = aws_get_region() )
```

**Arguments**
- **VpcId** Character. The ID of the VPC.
- **DryRun** Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]
- **TagSpecification** List. The tags to assign to the network ACL.[optional]
- **simplify** Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others** Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error** Logical. Whether to show an error message when a network error occurs.
- **retry_time** Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- **network_timeout** Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region** Character. The region of the AWS service.

**Value**
A list object or a character vector
VpcId

The ID of the VPC.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

TagSpecification

The tags to assign to the network ACL.

description

Create Network Acl Entry

Usage

c2_create_network_acl_entry(
    Egress,
    NetworkAclId,
    Protocol,
    RuleAction,
    RuleNumber,
    CidrBlock = NULL,
    DryRun = NULL,
    Icmp = NULL,
    Ipv6CidrBlock = NULL,
    PortRange = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

Egress Logical. Indicates whether this is an egress rule (rule is applied to traffic leaving the subnet).

NetworkAclId Character. The ID of the network ACL.
Protocol    Character. The protocol number.
RuleAction  Character. Indicates whether to allow or deny the traffic that matches the rule.
RuleNumber  Integer. The rule number for the entry (for example, 100).
CidrBlock   Character. The IPv4 network range to allow or deny, in CIDR notation (for example 172.16.0.0/24). We modify...[optional]
DryRun      Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]
Icmp        Object. ICMP protocol: The ICMP or ICMPv6 type and code.[optional]
Ipv6CidrBlock Character. The IPv6 network range to allow or deny, in CIDR notation (for example 2001:db8:1234:1a00::/64).[optional]
PortRange   Object. TCP or UDP protocols: The range of ports the rule applies to.[optional]
simplify    Logical. Whether to simplify the result and handle nextToken in the response[optional]
others      Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region      Character. The region of the AWS service.

Value

A list object or a character vector

Egress

Indicates whether this is an egress rule (rule is applied to traffic leaving the subnet).

NetworkAclId

The ID of the network ACL.

Protocol

The protocol number. A value of '-1' means all protocols. If you specify '-1' or a protocol number other than '-6' (TCP), '-17' (UDP), or '-1' (ICMP), traffic on all ports is allowed, regardless of any ports or ICMP types or codes that you specify. If you specify protocol '-58' (ICMPv6) and specify an IPv4 CIDR block, traffic for all ICMP types and codes allowed, regardless of any that you specify. If you specify protocol '-58' (ICMPv6) and specify an IPv6 CIDR block, you must specify an ICMP type and code.

RuleAction

Indicates whether to allow or deny the traffic that matches the rule.
RuleNumber

The rule number for the entry (for example, 100). ACL entries are processed in ascending order by rule number.

Constraints: Positive integer from 1 to 32766. The range 32767 to 65535 is reserved for internal use.

CidrBlock

The IPv4 network range to allow or deny, in CIDR notation (for example 172.16.0.0/24). We modify the specified CIDR block to its canonical form; for example, if you specify 100.68.0.18/18, we modify it to 100.68.0.0/18.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Icmp

ICMP protocol: The ICMP or ICMPv6 type and code. Required if specifying protocol 1 (ICMP) or protocol 58 (ICMPv6) with an IPv6 CIDR block.

Ipv6CidrBlock

The IPv6 network range to allow or deny, in CIDR notation (for example 2001:db8:1234:1a00::/64).

PortRange

TCP or UDP protocols: The range of ports the rule applies to. Required if specifying protocol 6 (TCP) or 17 (UDP).

Create Network Insights Path

Create Network Insights Path
Usage

```r
e2_create_network_insights_path(
  Source,
  Destination,
  Protocol,
  ClientToken,
  SourceIp = NULL,
  DestinationIp = NULL,
  DestinationPort = NULL,
  TagSpecification = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **Source** Character. The AWS resource that is the source of the path.
- **Destination** Character. The AWS resource that is the destination of the path.
- **Protocol** Character. The protocol.
- **ClientToken** Character. Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.
- **SourceIp** Character. The IP address of the AWS resource that is the source of the path.[optional]
- **DestinationIp** Character. The IP address of the AWS resource that is the destination of the path.[optional]
- **DestinationPort** Integer. The destination port.[optional]
- **TagSpecification** List. The tags to add to the path.[optional]
- **DryRun** Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
- **simplify** Logical. Whether to simplify the result and handle `nextToken` in the response[optional]
- **others** Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error** Logical. Whether to show an error message when a network error occurs.
- **retry_time** Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout** Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region** Character. The region of the AWS service.
Value

A list object or a character vector

Source

The AWS resource that is the source of the path.

Destination

The AWS resource that is the destination of the path.

Protocol

The protocol.

ClientToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. For more information, see How to Ensure Idempotency.

SourceIp

The IP address of the AWS resource that is the source of the path.

DestinationIp

The IP address of the AWS resource that is the destination of the path.

DestinationPort

The destination port.

TagSpecification

The tags to add to the path.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
ec2_create_network_interface

Create Network Interface

Description

Create Network Interface

Usage

ect2_create_network_interface(
    SubnetId,
    Description = NULL,
    DryRun = NULL,
    SecurityGroupId = NULL,
    Ipv6AddressCount = NULL,
    Ipv6Addresses = NULL,
    PrivateIpAddress = NULL,
    PrivateIpAddresses = NULL,
    SecondaryPrivateIpAddressCount = NULL,
    InterfaceType = NULL,
    TagSpecification = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

SubnetId  Character. The ID of the subnet to associate with the network interface.
Description Character. A description for the network interface.[optional]
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
SecurityGroupId List. The IDs of one or more security groups.[optional]
Ipv6AddressCount Integer. The number of IPv6 addresses to assign to a network interface.[optional]
Ipv6Addresses List. One or more specific IPv6 addresses from the IPv6 CIDR block range of your subnet.[optional]
PrivateIpAddress Character. The primary private IPv4 address of the network interface.[optional]
**PrivateIpAddresses**
- List. One or more private IPv4 addresses.[optional]

**SecondaryPrivateIpAddressCount**
- Integer. The number of secondary private IPv4 addresses to assign to a network interface.[optional]

**InterfaceType**
- Character. Indicates the type of network interface.[optional]

**TagSpecification**
- List. The tags to apply to the new network interface.[optional]

**simplify**
- Logical. Whether to simplify the result and handle `nextToken` in the response[optional]

**others**
- Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

**print_on_error**
- Logical. Whether to show an error message when a network error occurs.

**retry_time**
- Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

**network_timeout**
- Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

**region**
- Character. The region of the AWS service.

**Value**
- A list object or a character vector

**SubnetId**
- The ID of the subnet to associate with the network interface.

**Description**
- A description for the network interface.

**DryRun**
- Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

**SecurityGroupId**
- The IDs of one or more security groups.

**Ipv6AddressCount**
- The number of IPv6 addresses to assign to a network interface. Amazon EC2 automatically selects the IPv6 addresses from the subnet range. You can't use this option if specifying specific IPv6 addresses. If your subnet has the `AssignIpv6AddressOnCreation` attribute set to true, you can specify 0 to override this setting.
**Ipv6Addresses**

One or more specific IPv6 addresses from the IPv6 CIDR block range of your subnet. You can’t use this option if you’re specifying a number of IPv6 addresses.

**PrivateIpAddress**

The primary private IPv4 address of the network interface. If you don’t specify an IPv4 address, Amazon EC2 selects one for you from the subnet’s IPv4 CIDR range. If you specify an IP address, you cannot indicate any IP addresses specified in privateIpAddresses as primary (only one IP address can be designated as primary).

**PrivateIpAddresses**

One or more private IPv4 addresses.

**SecondaryPrivateIpAddressCount**

The number of secondary private IPv4 addresses to assign to a network interface. When you specify a number of secondary IPv4 addresses, Amazon EC2 selects these IP addresses within the subnet’s IPv4 CIDR range. You can’t specify this option and specify more than one private IP address using privateIpAddresses.

The number of IP addresses you can assign to a network interface varies by instance type. For more information, see IP Addresses Per ENI Per Instance Type in the Amazon Virtual Private Cloud User Guide.

**InterfaceType**

Indicates the type of network interface. To create an Elastic Fabric Adapter (EFA), specify efa. For more information, see Elastic Fabric Adapter in the Amazon Elastic Compute Cloud User Guide.

**TagSpecification**

The tags to apply to the new network interface.
Usage

```r
ec2_create_network_interface_permission(
    NetworkInterfaceId,
    Permission,
    AwsAccountId = NULL,
    AwsService = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **NetworkInterfaceId**: Character. The ID of the network interface.
- **Permission**: Character. The type of permission to grant.
- **AwsAccountId**: Character. The AWS account ID.[optional]
- **AwsService**: Character. The AWS service. Currently not supported.[optional]
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
- **simplify**: Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

**NetworkInterfaceId**

The ID of the network interface.

**Permission**

The type of permission to grant.
**AwsAccountId**

The AWS account ID.

**AwsService**

The AWS service. Currently not supported.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**ec2_create_placement_group**

Create Placement Group

---

**Description**

Create Placement Group

**Usage**

```r
ec2_create_placement_group(
  DryRun = NULL,
  GroupName = NULL,
  Strategy = NULL,
  PartitionCount = NULL,
  TagSpecification = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- **DryRun** Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **GroupName** Character. A name for the placement group. [optional]
- **Strategy** Character. The placement strategy. [optional]
- **PartitionCount** Integer. The number of partitions. Valid only when **Strategy** is set to partition. [optional]
- **TagSpecification** List. The tags to apply to the new placement group. [optional]
```r
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value
A list object or a character vector

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

GroupName
A name for the placement group. Must be unique within the scope of your account for the Region.
Constraints: Up to 255 ASCII characters

Strategy
The placement strategy.

PartitionCount
The number of partitions. Valid only when Strategy is set to partition.

TagSpecification
The tags to apply to the new placement group.
```
ec2_create_reserved_instances_listing

Create Reserved Instances Listing

Description
Create Reserved Instances Listing

Usage
ec2_create_reserved_instances_listing(
  ClientToken,
  InstanceCount,
  PriceSchedules,
  ReservedInstancesId,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

ClientToken
  Character. Unique, case-sensitive identifier you provide to ensure idempotency of your listings.

InstanceCount
  Integer. The number of instances that are a part of a Reserved Instance account to be listed in the Reserved...

PriceSchedules
  List. A list specifying the price of the Standard Reserved Instance for each month remaining in the Reserved...

ReservedInstancesId
  Character. The ID of the active Standard Reserved Instance.

simplify
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
  Logical. Whether to show an error message when a network error occurs.

retry_time
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
  Character. The region of the AWS service.
**Value**

A list object or a character vector

**ClientToken**

Unique, case-sensitive identifier you provide to ensure idempotency of your listings. This helps avoid duplicate listings. For more information, see [Ensuring Idempotency](#).

**InstanceCount**

The number of instances that are a part of a Reserved Instance account to be listed in the Reserved Instance Marketplace. This number should be less than or equal to the instance count associated with the Reserved Instance ID specified in this call.

**PriceSchedules**

A list specifying the price of the Standard Reserved Instance for each month remaining in the Reserved Instance term.

**ReservedInstancesId**

The ID of the active Standard Reserved Instance.

---

**Usage**

```r
e2_create_route(  
    RouteTableId,  
    DestinationCidrBlock = NULL,  
    DestinationIpv6CidrBlock = NULL,  
    DestinationPrefixListId = NULL,  
    DryRun = NULL,  
    VpcEndpointId = NULL,  
    EgressOnlyInternetGatewayId = NULL,  
    GatewayId = NULL,  
    InstanceId = NULL,  
    NatGatewayId = NULL,  
    TransitGatewayId = NULL,  
    LocalGatewayId = NULL,  
    CarrierGatewayId = NULL,  
    NetworkInterfaceId = NULL,  
)```

**Description**

Create Route
VpcPeeringConnectionId = NULL,
simplify = TRUE,
others = list(),
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

RouteTableId Character. The ID of the route table for the route.
DestinationCidrBlock
Character. The IPv4 CIDR address block used for the destination match.[optional]
DestinationIpv6CidrBlock
Character. The IPv6 CIDR block used for the destination match.[optional]
DestinationPrefixListId
Character. The ID of a prefix list used for the destination match.[optional]
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
VpcEndpointId Character. The ID of a VPC endpoint. Supported for Gateway Load Balancer endpoints only.[optional]
EgressOnlyInternetGatewayId
Character. [IPv6 traffic only] The ID of an egress-only internet gateway.[optional]
GatewayId Character. The ID of an internet gateway or virtual private gateway attached to your VPC.[optional]
InstanceId Character. The ID of a NAT instance in your VPC.[optional]
NatGatewayId Character. [IPv4 traffic only] The ID of a NAT gateway.[optional]
TransitGatewayId
Character. The ID of a transit gateway.[optional]
LocalGatewayId Character. The ID of the local gateway.[optional]
CarrierGatewayId
Character. The ID of the carrier gateway.[optional]
NetworkInterfaceId
Character. The ID of a network interface.[optional]
VpcPeeringConnectionId
Character. The ID of a VPC peering connection.[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout
   Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
   Character. The region of the AWS service.

**Value**

A list object or a character vector

**RouteTableId**

The ID of the route table for the route.

**DestinationCidrBlock**

The IPv4 CIDR address block used for the destination match. Routing decisions are based on the most specific match. We modify the specified CIDR block to its canonical form; for example, if you specify `100.68.0.18/18`, we modify it to `100.68.0.0/18`.

**DestinationIpv6CidrBlock**

The IPv6 CIDR block used for the destination match. Routing decisions are based on the most specific match.

**DestinationPrefixListId**

The ID of a prefix list used for the destination match.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

**VpcEndpointId**

The ID of a VPC endpoint. Supported for Gateway Load Balancer endpoints only.

**EgressOnlyInternetGatewayId**

[IPv6 traffic only] The ID of an egress-only internet gateway.

**GatewayId**

The ID of an internet gateway or virtual private gateway attached to your VPC.

**InstanceId**

The ID of a NAT instance in your VPC. The operation fails if you specify an instance ID unless exactly one network interface is attached.
ec2_create_route_table

NatGatewayId

[IPv4 traffic only] The ID of a NAT gateway.

TransitGatewayId

The ID of a transit gateway.

LocalGatewayId

The ID of the local gateway.

CarrierGatewayId

The ID of the carrier gateway.

You can only use this option when the VPC contains a subnet which is associated with a Wavelength Zone.

NetworkInterfaceId

The ID of a network interface.

VpcPeeringConnectionId

The ID of a VPC peering connection.

---

Description

Create Route Table

Usage

c2_create_route_table(
    VpcId,
    DryRun = NULL,
    TagSpecification = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
Arguments

- **VpcId**
  Character. The ID of the VPC.

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

- **TagSpecification**
  List. The tags to assign to the route table.[optional]

- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  Character. The region of the AWS service.

Value

A list object or a character vector

**VpcId**

The ID of the VPC.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

**TagSpecification**

The tags to assign to the route table.

---

**ec2_create_security_group**

*Create Security Group*

**Description**

Create Security Group
Usage

ec2_create_security_group(
    GroupDescription,
    GroupName,
    VpcId = NULL,
    TagSpecification = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

GroupDescription
Character. A description for the security group.

GroupName
Character. The name of the security group.

VpcId
Character. [EC2-VPC] The ID of the VPC. Required for EC2-VPC.[optional]

TagSpecification
List. The tags to assign to the security group.[optional]

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value

A list object or a character vector

GroupDescription

A description for the security group. This is informational only.

Constraints: Up to 255 characters in length

Constraints for EC2-Classic: ASCII characters

Constraints for EC2-VPC: a-z, A-Z, 0-9, spaces, and _\-\/:\#\@\[\+\&\;\!\$\*
**GroupName**

The name of the security group.

Constraints: Up to 255 characters in length. Cannot start with sg-.

Constraints for EC2-Classic: ASCII characters.

Constraints for EC2-VPC: a-z, A-Z, 0-9, spaces, and_.-_/()#,@[1]+=.&!$*

**VpcId**

[EC2-VPC] The ID of the VPC. Required for EC2-VPC.

**TagSpecification**

The tags to assign to the security group.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

```r
ec2_create_snapshot

Description

Create Snapshot

Usage

e2_create_snapshot(
  VolumeId,
  Description = NULL,
  OutpostArn = NULL,
  TagSpecification = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VolumeId</td>
<td>Character. The ID of the EBS volume.</td>
</tr>
<tr>
<td>Description</td>
<td>Character. A description for the snapshot.[optional]</td>
</tr>
<tr>
<td>OutpostArn</td>
<td>Character. The Amazon Resource Name (ARN) of the AWS Outpost on which</td>
</tr>
<tr>
<td></td>
<td>to create a local snapshot.[optional]</td>
</tr>
<tr>
<td>TagSpecification</td>
<td>List. The tags to apply to the snapshot during creation.[optional]</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action,</td>
</tr>
<tr>
<td></td>
<td>without actually making the request,...[optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response[optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters</td>
</tr>
<tr>
<td></td>
<td>and need to be added into the request[optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network</td>
</tr>
<tr>
<td></td>
<td>issue. If the request has been sent retry_time times but still not be able</td>
</tr>
<tr>
<td></td>
<td>to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up.</td>
</tr>
<tr>
<td></td>
<td>Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>

Value

A list object or a character vector

VolumeId

The ID of the EBS volume.

Description

A description for the snapshot.

OutpostArn

The Amazon Resource Name (ARN) of the AWS Outpost on which to create a local snapshot.

- To create a snapshot of a volume in a Region, omit this parameter. The snapshot is created in the same Region as the volume.
- To create a snapshot of a volume on an Outpost and store the snapshot in the Region, omit this parameter. The snapshot is created in the Region for the Outpost.
- To create a snapshot of a volume on an Outpost and store the snapshot on an Outpost, specify the ARN of the destination Outpost. The snapshot must be created on the same Outpost as the volume.

For more information, see Creating local snapshots from volumes on an Outpost in the Amazon Elastic Compute Cloud User Guide.
TagSpecification

The tags to apply to the snapshot during creation.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_create_snapshots  Create Snapshots

Description

Create Snapshots

Usage

ec2_create_snapshots(
  InstanceSpecification,
  Description = NULL,
  OutpostArn = NULL,
  TagSpecification = NULL,
  DryRun = NULL,
  CopyTagsFromSource = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

InstanceSpecification  Object. The instance to specify which volumes should be included in the snapshots.

Description  Character. A description propagated to every snapshot specified by the instance.[optional]

OutpostArn  Character. The Amazon Resource Name (ARN) of the AWS Outpost on which to create the local snapshots.[optional]

TagSpecification  List. Tags to apply to every snapshot specified by the instance.[optional]

DryRun  Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]
CopyTagsFromSource
  Character. Copies the tags from the specified volume to corresponding snapshot.[optional]

simplify
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
  Logical. Whether to show an error message when a network error occurs.

retry_time
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
  Character. The region of the AWS service.

Value
  A list object or a character vector

InstanceSpecification
  The instance to specify which volumes should be included in the snapshots.

Description
  A description propagated to every snapshot specified by the instance.

OutpostArn
  The Amazon Resource Name (ARN) of the AWS Outpost on which to create the local snapshots.

  • To create snapshots from an instance in a Region, omit this parameter. The snapshots are created in the same Region as the instance.
  • To create snapshots from an instance on an Outpost and store the snapshots in the Region, omit this parameter. The snapshots are created in the Region for the Outpost.
  • To create snapshots from an instance on an Outpost and store the snapshots on an Outpost, specify the ARN of the destination Outpost. The snapshots must be created on the same Outpost as the instance.

  For more information, see Creating multi-volume local snapshots from instances on an Outpost in the Amazon Elastic Compute Cloud User Guide.

TagSpecification
  Tags to apply to every snapshot specified by the instance.
**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

**CopyTagsFromSource**

Copies the tags from the specified volume to corresponding snapshot.

---

**Description**

Creates a data feed for Spot Instances, enabling you to view Spot Instance usage logs. You can create one data feed per AWS account. For more information, see Spot Instance data feed in the Amazon EC2 User Guide for Linux Instances.

**Usage**

```r
ec2_create_spot_datafeed_subscription(
  Bucket,
  DryRun = NULL,
  Prefix = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- **Bucket**: Character. The name of the Amazon S3 bucket in which to store the Spot Instance data feed.
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
- **Prefix**: Character. The prefix for the data feed file names.[optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response[optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
**retry_time**
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

**network_timeout**
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

**region**
Character. The region of the AWS service.

**Value**
A list object or a character vector

**Bucket**
The name of the Amazon S3 bucket in which to store the Spot Instance data feed. For more information about bucket names, see [Rules for bucket naming](https://docs.aws.amazon.com/AmazonS3/latest/userguide/BucketNaming.html) in the Amazon S3 Developer Guide.

**DryRun**
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

**Prefix**
The prefix for the data feed file names.

---

**Description**
Create Subnet

**Usage**
```r
e2.create_subnet(
  VpcId,
  CidrBlock,
  TagSpecification = NULL,
  AvailabilityZone = NULL,
  AvailabilityZoneId = NULL,
  Ipv6CidrBlock = NULL,
  OutpostArn = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
)```
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

**Arguments**

- **VpcId**  
  Character. The ID of the VPC.
- **CidrBlock**  
  Character. The IPv4 network range for the subnet, in CIDR notation.
- **TagSpecification**  
  List. The tags to assign to the subnet.[optional]
- **AvailabilityZone**  
  Character. The Availability Zone or Local Zone for the subnet.[optional]
- **AvailabilityZoneId**  
  Character. The AZ ID or the Local Zone ID of the subnet.[optional]
- **Ipv6CidrBlock**  
  Character. The IPv6 network range for the subnet, in CIDR notation.[optional]
- **OutpostArn**  
  Character. The Amazon Resource Name (ARN) of the Outpost.[optional]
- **DryRun**  
  Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
- **simplify**  
  Logical. Whether to simplify the result and handle nextToken in the response.[optional]
- **others**  
  Named list. The parameters that are not included in the function parameters and need to be added into the request.[optional]
- **print_on_error**  
  Logical. Whether to show an error message when a network error occurs.
- **retry_time**  
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- **network_timeout**  
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**  
  Character. The region of the AWS service.

**Value**

A list object or a character vector

**VpcId**

The ID of the VPC.

**CidrBlock**

The IPv4 network range for the subnet, in CIDR notation. For example, 10.0.0.0/24. We modify the specified CIDR block to its canonical form; for example, if you specify 100.68.0.18/18, we modify it to 100.68.0.0/18.
TagSpecification

The tags to assign to the subnet.

AvailabilityZone

The Availability Zone or Local Zone for the subnet.

Default: AWS selects one for you. If you create more than one subnet in your VPC, we do not necessarily select a different zone for each subnet.

To create a subnet in a Local Zone, set this value to the Local Zone ID, for example us-west-2-lax-1a. For information about the Regions that support Local Zones, see Available Regions in the Amazon Elastic Compute Cloud User Guide.

To create a subnet in an Outpost, set this value to the Availability Zone for the Outpost and specify the Outpost ARN.

AvailabilityZoneId

The AZ ID or the Local Zone ID of the subnet.

Ipv6CidrBlock

The IPv6 network range for the subnet, in CIDR notation. The subnet size must use a /64 prefix length.

OutpostArn

The Amazon Resource Name (ARN) of the Outpost. If you specify an Outpost ARN, you must also specify the Availability Zone of the Outpost subnet.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**ec2_create_tags**

**Create Tags**

**Description**

Create Tags
**Usage**

```r
ec2_create_tags(
    ResourceId,
    Tag,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResourceId</td>
<td>List. The IDs of the resources, separated by spaces.</td>
</tr>
<tr>
<td>Tag</td>
<td>List. The tags.</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response. [optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>

**Value**

A list object or a character vector

**ResourceId**

The IDs of the resources, separated by spaces.

Constraints: Up to 1000 resource IDs. We recommend breaking up this request into smaller batches.

**Tag**

The tags. The value parameter is required, but if you don’t want the tag to have a value, specify the parameter with no value, and we set the value to an empty string.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**Description**

Create Traffic Mirror Filter

**Usage**

```r
ec2_create_traffic_mirror_filter(
    Description = NULL,
    TagSpecification = NULL,
    DryRun = NULL,
    ClientToken = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Character. The description of the Traffic Mirror filter.[optional]</td>
</tr>
<tr>
<td>TagSpecification</td>
<td>List. The tags to assign to a Traffic Mirror filter.[optional]</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]</td>
</tr>
<tr>
<td>ClientToken</td>
<td>Character. Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.[optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response.[optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request.[optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
</tr>
</tbody>
</table>
network_timeout
    Numeric. Number of seconds to wait for a REST response until giving up. Can
    not be less than 1 ms.

region
    Character. The region of the AWS service.

Value
    A list object or a character vector

Description
    The description of the Traffic Mirror filter.

TagSpecification
    The tags to assign to a Traffic Mirror filter.

DryRun
    Checks whether you have the required permissions for the action, without actually making the
    request, and provides an error response. If you have the required permissions, the error response is
    DryRunOperation. Otherwise, it is UnauthorizedOperation.

ClientToken
    Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. For
    more information, see How to Ensure Idempotency.

---

ec2_create_traffic_mirror_filter_rule

Create Traffic Mirror Filter Rule

---

Description

Create Traffic Mirror Filter Rule

Usage

ec2_create_traffic_mirror_filter_rule(
    TrafficMirrorFilterId,
    TrafficDirection,
    RuleNumber,
    RuleAction, 
    DestinationCidrBlock, 
    SourceCidrBlock, 
    DestinationPortRange = NULL, 
    SourcePortRange = NULL, 
    Protocol = NULL,
ec2_create_traffic_mirror_filter_rule

```r
Description = NULL,
DryRun = NULL,
ClientToken = NULL,
simplify = TRUE,
others = list(),
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
```

### Arguments

- **TrafficMirrorFilterId**
  - Character. The ID of the filter that this rule is associated with.

- **TrafficDirection**
  - Character. The type of traffic (ingress | egress).

- **RuleNumber**
  - Integer. The number of the Traffic Mirror rule.

- **RuleAction**
  - Character. The action to take (accept | reject) on the filtered traffic.

- **DestinationCidrBlock**
  - Character. The destination CIDR block to assign to the Traffic Mirror rule.

- **SourceCidrBlock**
  - Character. The source CIDR block to assign to the Traffic Mirror rule.

- **DestinationPortRange**
  - Object. The destination port range.[optional]

- **SourcePortRange**
  - Object. The source port range.[optional]

- **Protocol**
  - Integer. The protocol, for example UDP, to assign to the Traffic Mirror rule.[optional]

- **Description**
  - Character. The description of the Traffic Mirror rule.[optional]

- **DryRun**
  - Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]

- **ClientToken**
  - Character. Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.[optional]

- **simplify**
  - Logical. Whether to simplify the result and handle nextToken in the response[optional]

- **others**
  - Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  - Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  - Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  - Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  - Character. The region of the AWS service.
Value
A list object or a character vector

TrafficMirrorFilterId
The ID of the filter that this rule is associated with.

TrafficDirection
The type of traffic (ingress \| egress).

RuleNumber
The number of the Traffic Mirror rule. This number must be unique for each Traffic Mirror rule in a given direction. The rules are processed in ascending order by rule number.

RuleAction
The action to take (accept \| reject) on the filtered traffic.

DestinationCidrBlock
The destination CIDR block to assign to the Traffic Mirror rule.

SourceCidrBlock
The source CIDR block to assign to the Traffic Mirror rule.

DestinationPortRange
The destination port range.

SourcePortRange
The source port range.

Protocol
The protocol, for example UDP, to assign to the Traffic Mirror rule.
For information about the protocol value, see Protocol Numbers on the Internet Assigned Numbers Authority (IANA) website.

Description
The description of the Traffic Mirror rule.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
ClientToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. For more information, see How to Ensure Idempotency.

Description

Create Traffic Mirror Session

Usage

```
ec2_create_traffic_mirror_session(
    NetworkInterfaceId,
    TrafficMirrorTargetId,
    TrafficMirrorFilterId,
    SessionNumber,
    PacketLength = NULL,
    VirtualNetworkId = NULL,
    Description = NULL,
    TagSpecification = NULL,
    DryRun = NULL,
    ClientToken = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

NetworkInterfaceId

Character. The ID of the source network interface.

TrafficMirrorTargetId

Character. The ID of the Traffic Mirror target.

TrafficMirrorFilterId

Character. The ID of the Traffic Mirror filter.

SessionNumber

Integer. The session number determines the order in which sessions are evaluated when an interface is used...

PacketLength

Integer. The number of bytes in each packet to mirror.[optional]

VirtualNetworkId

Integer. The VXLAN ID for the Traffic Mirror session.[optional]
Description
Character. The description of the Traffic Mirror session.[optional]

TagSpecification
List. The tags to assign to a Traffic Mirror session.[optional]

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

ClientToken
Character. Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value
A list object or a character vector

NetworkInterfaceId
The ID of the source network interface.

TrafficMirrorTargetId
The ID of the Traffic Mirror target.

TrafficMirrorFilterId
The ID of the Traffic Mirror filter.

SessionNumber
The session number determines the order in which sessions are evaluated when an interface is used by multiple sessions. The first session with a matching filter is the one that mirrors the packets. Valid values are 1-32766.
PacketLength

The number of bytes in each packet to mirror. These are bytes after the VXLAN header. Do not specify this parameter when you want to mirror the entire packet. To mirror a subset of the packet, set this to the length (in bytes) that you want to mirror. For example, if you set this value to 100, then the first 100 bytes that meet the filter criteria are copied to the target.

If you do not want to mirror the entire packet, use the PacketLength parameter to specify the number of bytes in each packet to mirror.

VirtualNetworkId

The VXLAN ID for the Traffic Mirror session. For more information about the VXLAN protocol, see RFC 7348. If you do not specify a VirtualNetworkId, an account-wide unique id is chosen at random.

Description

The description of the Traffic Mirror session.

TagSpecification

The tags to assign to a Traffic Mirror session.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ClientToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. For more information, see How to Ensure Idempotency.
Usage

```r
e2_create_traffic_mirror_target(
    NetworkInterfaceId = NULL,
    NetworkLoadBalancerArn = NULL,
    Description = NULL,
    TagSpecification = NULL,
    DryRun = NULL,
    ClientToken = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **NetworkInterfaceId**
  Character. The network interface ID that is associated with the target. [optional]

- **NetworkLoadBalancerArn**
  Character. The Amazon Resource Name (ARN) of the Network Load Balancer that is associated with the target. [optional]

- **Description**
  Character. The description of the Traffic Mirror target. [optional]

- **TagSpecification**
  List. The tags to assign to the Traffic Mirror target. [optional]

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]

- **ClientToken**
  Character. Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. [optional]

- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encountering the network issue. If the request has been sent `retry_time` times but still not able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.

- **region**
  Character. The region of the AWS service.

Value

A list object or a character vector
NetworkInterfaceId
The network interface ID that is associated with the target.

NetworkLoadBalancerArn
The Amazon Resource Name (ARN) of the Network Load Balancer that is associated with the target.

Description
The description of the Traffic Mirror target.

TagSpecification
The tags to assign to the Traffic Mirror target.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ClientToken
Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. For more information, see How to Ensure Idempotency.

Usage

```python
e2_create_transit_gateway(
    Description = NULL,
    Options = NULL,
    TagSpecification = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```
Arguments

- **Description**: Character. A description of the transit gateway. [optional]
- **Options**: Object. The transit gateway options. [optional]
- **TagSpecification**: List. The tags to apply to the transit gateway. [optional]
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

Description

A description of the transit gateway.

Options

The transit gateway options.

TagSpecification

The tags to apply to the transit gateway.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is **DryRunOperation**. Otherwise, it is **UnauthorizedOperation**.
Description

Create Transit Gateway Connect

Usage

```r
ec2_create_transit_gateway_connect(
  TransportTransitGatewayAttachmentId,
  Options,
  TagSpecification = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **TransportTransitGatewayAttachmentId**
  Character. The ID of the transit gateway attachment.

- **Options**
  Object. The Connect attachment options.

- **TagSpecification**
  List. The tags to apply to the Connect attachment.[optional]

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  Character. The region of the AWS service.
ec2_create_transit_gateway_connect_peer

Value
A list object or a character vector

TransportTransitGatewayAttachmentId
The ID of the transit gateway attachment. You can specify a VPC attachment or a AWS Direct Connect attachment.

Options
The Connect attachment options.

TagSpecification
The tags to apply to the Connect attachment.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

description
Create Transit Gateway Connect Peer

Usage
```r
ec2_create_transit_gateway_connect_peer(
  TransitGatewayAttachmentId, 
  PeerAddress, 
  InsideCidrBlocks, 
  TransitGatewayAddress = NULL, 
  BgpOptions = NULL, 
  TagSpecification = NULL, 
  DryRun = NULL, 
  simplify = TRUE, 
  others = list(), 
  print_on_error = aws_get_print_on_error(), 
  retry_time = aws_get_retry_time(), 
  network_timeout = aws_get_network_timeout(), 
  region = aws_get_region()
)
```
Arguments

TransitGatewayAttachmentId
  Character. The ID of the Connect attachment.

PeerAddress
  Character. The peer IP address (GRE outer IP address) on the appliance side of
  the Connect peer.

InsideCidrBlocks
  List. The range of inside IP addresses that are used for BGP peering.

TransitGatewayAddress
  Character. The peer IP address (GRE outer IP address) on the transit gateway
  side of the Connect peer, which...[optional]

BgpOptions
  Object. The BGP options for the Connect peer.[optional]

TagSpecification
  List. The tags to apply to the Connect peer.[optional]

DryRun
  Logical. Checks whether you have the required permissions for the action, with-
  out actually making the request...,.[optional]

simplify
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
  Named list. The parameters that are not included in the function parameters and
  need to be added into the request[optional]

print_on_error
  Logical. Whether to show an error message when a network error occurs.

retry_time
  Integer. Number of retries for a REST request when encounter the network
  issue. If the request has been sent retry_time times but still not be able to get
  the response, an error will be thrown.

network_timeout
  Numeric. Number of seconds to wait for a REST response until giving up. Can
  not be less than 1 ms.

region
  Character. The region of the AWS service.

Value

A list object or a character vector

TransitGatewayAttachmentId

The ID of the Connect attachment.

PeerAddress

The peer IP address (GRE outer IP address) on the appliance side of the Connect peer.

InsideCidrBlocks

The range of inside IP addresses that are used for BGP peering. You must specify a size /29 IPv4
CIDR block from the 169.254.0.0/16 range. The first address from the range must be configured
on the appliance as the BGP IP address. You can also optionally specify a size /125 IPv6 CIDR
block from the fd00::/8 range.
TransitGatewayAddress

The peer IP address (GRE outer IP address) on the transit gateway side of the Connect peer, which must be specified from a transit gateway CIDR block. If not specified, Amazon automatically assigns the first available IP address from the transit gateway CIDR block.

BgpOptions

The BGP options for the Connect peer.

TagSpecification

The tags to apply to the Connect peer.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
Arguments

TransitGatewayId
Character. The ID of the transit gateway.

Options
Object. The options for the transit gateway multicast domain.[optional]

TagSpecification
List. The tags for the transit gateway multicast domain.[optional]

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value
A list object or a character vector

TransitGatewayId
The ID of the transit gateway.

Options
The options for the transit gateway multicast domain.

TagSpecification
The tags for the transit gateway multicast domain.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
ec2_create_transit_gateway_peering_attachment

Create Transit Gateway Peering Attachment

Description
Create Transit Gateway Peering Attachment

Usage
ec2_create_transit_gateway_peering_attachment(
    TransitGatewayId,
    PeerTransitGatewayId,
    PeerAccountId,
    PeerRegion,
    TagSpecification = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments
TransitGatewayId
    Character. The ID of the transit gateway.

PeerTransitGatewayId
    Character. The ID of the peer transit gateway with which to create the peering
    attachment.

PeerAccountId
    Character. The AWS account ID of the owner of the peer transit gateway.

PeerRegion
    Character. The Region where the peer transit gateway is located.

TagSpecification
    List. The tags to apply to the transit gateway peering attachment.[optional]

DryRun
    Logical. Checks whether you have the required permissions for the action, with-
    out actually making the request....[optional]

simplify
    Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
    Named list. The parameters that are not included in the function parameters and
    need to be added into the request[optional]

print_on_error
    Logical. Whether to show an error message when a network error occurs.

retry_time
    Integer. Number of retries for a REST request when encounter the network
    issue. If the request has been sent retry_time times but still not be able to get
    the response, an error will be thrown.
**network_timeout**

Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.

**region**

Character. The region of the AWS service.

**Value**

A list object or a character vector

**TransitGatewayId**

The ID of the transit gateway.

**PeerTransitGatewayId**

The ID of the peer transit gateway with which to create the peering attachment.

**PeerAccountId**

The AWS account ID of the owner of the peer transit gateway.

**PeerRegion**

The Region where the peer transit gateway is located.

**TagSpecification**

The tags to apply to the transit gateway peering attachment.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**Description**

Creates a reference (route) to a prefix list in a specified transit gateway route table.
Usage

e2_create_transit_gateway_prefix_list_reference(
    TransitGatewayRouteTableId,
    PrefixListId,
    TransitGatewayAttachmentId = NULL,
    Blackhole = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

TransitGatewayRouteTableId
    Character. The ID of the transit gateway route table.

PrefixListId
    Character. The ID of the prefix list that is used for destination matches.

TransitGatewayAttachmentId
    Character. The ID of the attachment to which traffic is routed. [optional]

Blackhole
    Logical. Indicates whether to drop traffic that matches this route. [optional]

DryRun
    Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]

simplify
    Logical. Whether to simplify the result and handle nextToken in the response [optional]

others
    Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]

print_on_error
    Logical. Whether to show an error message when a network error occurs.

retry_time
    Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
    Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
    Character. The region of the AWS service.

Value

A list object or a character vector

TransitGatewayRouteTableId

The ID of the transit gateway route table.
PrefixListId

The ID of the prefix list that is used for destination matches.

TransitGatewayAttachmentId

The ID of the attachment to which traffic is routed.

Blackhole

Indicates whether to drop traffic that matches this route.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description

Creates a static route for the specified transit gateway route table.

Usage

c2.create_transit_gateway_route(
    DestinationCidrBlock,
    TransitGatewayRouteTableId,
    TransitGatewayAttachmentId = NULL,
    Blackhole = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws.get_print_on_error(),
    retry_time = aws.get_retry_time(),
    network_timeout = aws.get_network_timeout(),
    region = aws.get_region()
)

Arguments

DestinationCidrBlock
    Character. The CIDR range used for destination matches.

TransitGatewayRouteTableId
    Character. The ID of the transit gateway route table.
TransitGatewayAttachmentId
  Character. The ID of the attachment.[optional]

Blackhole
  Logical. Indicates whether to drop traffic that matches this route.[optional]

DryRun
  Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]

simplify
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
  Logical. Whether to show an error message when a network error occurs.

retry_time
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
  Character. The region of the AWS service.

Value
  A list object or a character vector

DestinationCidrBlock
  The CIDR range used for destination matches. Routing decisions are based on the most specific match.

TransitGatewayRouteTableId
  The ID of the transit gateway route table.

TransitGatewayAttachmentId
  The ID of the attachment.

Blackhole
  Indicates whether to drop traffic that matches this route.

DryRun
  Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
ec2_create_transit_gateway_route_table

Create Transit Gateway Route Table

Description
Creates a route table for the specified transit gateway.

Usage
ec2_create_transit_gateway_route_table(
    TransitGatewayId,
    TagSpecifications = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments
TransitGatewayId
    Character. The ID of the transit gateway.
TagSpecifications
    List. The tags to apply to the transit gateway route table.[optional]
DryRun
    Logical. Checks whether you have the required permissions for the action, with- out actually making the request.[optional]
simplify
    Logical. Whether to simplify the result and handle nextToken in the response[optional]
others
    Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error
    Logical. Whether to show an error message when a network error occurs.
retry_time
    Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout
    Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region
    Character. The region of the AWS service.

Value
A list object or a character vector
TransitGatewayId

The ID of the transit gateway.

TagSpecifications

The tags to apply to the transit gateway route table.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

describe Availability Zones

Create Transit Gateway Vpc Attachment

Create Transit Gateway Vpc Attachment

Usage

describe Availability Zones(  
    TransitGatewayId,  
    VpcId,  
    SubnetIds,  
    Options = NULL,  
    TagSpecifications = NULL,  
    DryRun = NULL,  
    simplify = TRUE,  
    others = list(),  
    print_on_error = aws_get_print_on_error(),  
    retry_time = aws_get_retry_time(),  
    network_timeout = aws_get_network_timeout(),  
    region = aws_get_region()
)

Arguments

TransitGatewayId

Character. The ID of the transit gateway.

VpcId

Character. The ID of the VPC.

SubnetIds

List. The IDs of one or more subnets.

Options

Object. The VPC attachment options.[optional]
TagSpecifications

List. The tags to apply to the VPC attachment.[optional]

DryRun

Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

simplify

Logical. Whether to simplify the result and handle nextToken in the response[optional]

others

Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error

Logical. Whether to show an error message when a network error occurs.

retry_time

Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout

Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region

Character. The region of the AWS service.

Value

A list object or a character vector

TransitGatewayId

The ID of the transit gateway.

VpcId

The ID of the VPC.

SubnetIds

The IDs of one or more subnets. You can specify only one subnet per Availability Zone. You must specify at least one subnet, but we recommend that you specify two subnets for better availability. The transit gateway uses one IP address from each specified subnet.

Options

The VPC attachment options.

TagSpecifications

The tags to apply to the VPC attachment.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
**ec2_create_volume**

**Create Volume**

**Description**

Create Volume

**Usage**

```python
ec2_create_volume(
    AvailabilityZone,
    Encrypted = NULL,
    Iops = NULL,
    KmsKeyId = NULL,
    OutpostArn = NULL,
    Size = NULL,
    SnapshotId = NULL,
    VolumeType = NULL,
    DryRun = NULL,
    TagSpecification = NULL,
    MultiAttachEnabled = NULL,
    Throughput = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

**Arguments**

- **AvailabilityZone**
  Character. The Availability Zone in which to create the volume.

- **Encrypted**
  Logical. Indicates whether the volume should be encrypted.[optional]

- **Iops**
  Integer. The number of I/O operations per second (IOPS).[optional]

- **KmsKeyId**
  Character. The identifier of the AWS Key Management Service (AWS KMS) customer master key (CMK) to use for...[optional]

- **OutpostArn**
  Character. The Amazon Resource Name (ARN) of the Outpost.[optional]

- **Size**
  Integer. The size of the volume, in GiBs.[optional]

- **SnapshotId**
  Character. The snapshot from which to create the volume.[optional]

- **VolumeType**
  Character. The volume type.[optional]

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
TagSpecification

List. The tags to apply to the volume during creation.[optional]

MultiAttachEnabled

Logical. Indicates whether to enable Amazon EBS Multi-Attach.[optional]

Throughput

Integer. The throughput to provision for a volume, with a maximum of 1,000 MiB/s.[optional]

simplify

Logical. Whether to simplify the result and handle nextToken in the response[optional]

others

Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error

Logical. Whether to show an error message when a network error occurs.

retry_time

Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout

Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region

Character. The region of the AWS service.

Value

A list object or a character vector

AvailabilityZone

The Availability Zone in which to create the volume.

Encrypted

Indicates whether the volume should be encrypted. The effect of setting the encryption state to true depends on the volume origin (new or from a snapshot), starting encryption state, ownership, and whether encryption by default is enabled. For more information, see Encryption by default in the Amazon Elastic Compute Cloud User Guide.

Encrypted Amazon EBS volumes must be attached to instances that support Amazon EBS encryption. For more information, see Supported instance types.

Iops

The number of I/O operations per second (IOPS). For gp3, io1, and io2 volumes, this represents the number of IOPS that are provisioned for the volume. For gp2 volumes, this represents the baseline performance of the volume and the rate at which the volume accumulates I/O credits for bursting.

The following are the supported values for each volume type:

- gp3: 3,000-16,000 IOPS
- io1: 100-64,000 IOPS
- io2: 100-64,000 IOPS
For io1 and io2 volumes, we guarantee 64,000 IOPS only for Instances built on the Nitro System. Other instance families guarantee performance up to 32,000 IOPS.

This parameter is required for io1 and io2 volumes. The default for gp3 volumes is 3,000 IOPS. This parameter is not supported for gp2, st1, sc1, or standard volumes.

**KmsKeyId**

The identifier of the AWS Key Management Service (AWS KMS) customer master key (CMK) to use for Amazon EBS encryption. If this parameter is not specified, your AWS managed CMK for EBS is used. If KmsKeyId is specified, the encrypted state must be true.

You can specify the CMK using any of the following:

- Key ID. For example, 1234abcd-12ab-34cd-56ef-1234567890ab.
- Key alias. For example, alias/ExampleAlias.
- Key ARN. For example, arn:aws:kms:us-east-1:012345678910:key/1234abcd-12ab-34cd-56ef-1234567890ab.
- Alias ARN. For example, arn:aws:kms:us-east-1:012345678910:alias/ExampleAlias.

AWS authenticates the CMK asynchronously. Therefore, if you specify an ID, alias, or ARN that is not valid, the action can appear to complete, but eventually fails.

**OutpostArn**

The Amazon Resource Name (ARN) of the Outpost.

**Size**

The size of the volume, in GiBs. You must specify either a snapshot ID or a volume size. If you specify a snapshot, the default is the snapshot size. You can specify a volume size that is equal to or larger than the snapshot size.

The following are the supported volumes sizes for each volume type:

- gp2 and gp3: 1-16,384
- io1 and io2: 4-16,384
- st1 and sc1: 125-16,384
- standard: 1-1,024

**SnapshotId**

The snapshot from which to create the volume. You must specify either a snapshot ID or a volume size.
VolumeType

The volume type. This parameter can be one of the following values:

- General Purpose SSD: gp2 \ gp3
- Provisioned IOPS SSD: io1 \ io2
- Throughput Optimized HDD: st1
- Cold HDD: sc1
- Magnetic: standard

For more information, see Amazon EBS volume types in the Amazon Elastic Compute Cloud User Guide.

Default: gp2

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

TagSpecification

The tags to apply to the volume during creation.

MultiAttachEnabled

Indicates whether to enable Amazon EBS Multi-Attach. If you enable Multi-Attach, you can attach the volume to up to 16 Instances built on the Nitro System in the same Availability Zone. This parameter is supported with io1 and io2 volumes only. For more information, see Amazon EBS Multi-Attach in the Amazon Elastic Compute Cloud User Guide.

Throughput

The throughput to provision for a volume, with a maximum of 1,000 MiB/s.

This parameter is valid only for gp3 volumes.

Valid Range: Minimum value of 125. Maximum value of 1000.

Description

Create Vpc
Usage

```r
ec2_create_vpc(
    CidrBlock,
    AmazonProvidedIpv6CidrBlock = NULL,
    Ipv6Pool = NULL,
    Ipv6CidrBlock = NULL,
    DryRun = NULL,
    InstanceTenancy = NULL,
    Ipv6CidrBlockNetworkBorderGroup = NULL,
    TagSpecification = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- `CidrBlock` Character. The IPv4 network range for the VPC, in CIDR notation.
- `AmazonProvidedIpv6CidrBlock` Logical. Requests an Amazon-provided IPv6 CIDR block with a /56 prefix length for the VPC.[optional]
- `Ipv6Pool` Character. The ID of an IPv6 address pool from which to allocate the IPv6 CIDR block.[optional]
- `Ipv6CidrBlock` Character. The IPv6 CIDR block from the IPv6 address pool.[optional]
- `DryRun` Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]
- `InstanceTenancy` Character. The tenancy options for instances launched into the VPC.[optional]
- `Ipv6CidrBlockNetworkBorderGroup` Character. The name of the location from which we advertise the IPV6 CIDR block.[optional]
- `TagSpecification` List. The tags to assign to the VPC.[optional]
- `simplify` Logical. Whether to simplify the result and handle nextToken in the response[optional]
- `others` Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- `print_on_error` Logical. Whether to show an error message when a network error occurs.
- `retry_time` Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- `network_timeout` Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region

Character. The region of the AWS service.

Value

A list object or a character vector

CidrBlock

The IPv4 network range for the VPC, in CIDR notation. For example, 10.0.0.0/16. We modify the specified CIDR block to its canonical form; for example, if you specify 100.68.0.18/18, we modify it to 100.68.0.0/18.

AmazonProvidedIpv6CidrBlock

Requests an Amazon-provided IPv6 CIDR block with a /56 prefix length for the VPC. You cannot specify the range of IP addresses, or the size of the CIDR block.

Ipv6Pool

The ID of an IPv6 address pool from which to allocate the IPv6 CIDR block.

Ipv6CidrBlock

The IPv6 CIDR block from the IPv6 address pool. You must also specify Ipv6Pool in the request.

To let Amazon choose the IPv6 CIDR block for you, omit this parameter.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

InstanceTenancy

The tenancy options for instances launched into the VPC. For default, instances are launched with shared tenancy by default. You can launch instances with any tenancy into a shared tenancy VPC. For dedicated, instances are launched as dedicated tenancy instances by default. You can only launch instances with a tenancy of dedicated or host into a dedicated tenancy VPC.

Important: The host value cannot be used with this parameter. Use the default or dedicated values only.

Default: default

Ipv6CidrBlockNetworkBorderGroup

The name of the location from which we advertise the IPV6 CIDR block. Use this parameter to limit the address to this location.

You must set AmazonProvidedIpv6CidrBlock to true to use this parameter.
TagSpecification

The tags to assign to the VPC.

Description

Create Vpc Endpoint

Usage

```r
e2_create_vpc_endpoint(
  VpcId, 
  ServiceName, 
 DryRun = NULL, 
  VpcEndpointType = NULL, 
  PolicyDocument = NULL, 
  RouteTableId = NULL, 
  SubnetId = NULL, 
  SecurityGroupId = NULL, 
  ClientToken = NULL, 
  PrivateDnsEnabled = NULL, 
  TagSpecification = NULL, 
simplify = TRUE, 
others = list(), 
print_on_error = aws_get_print_on_error(), 
retry_time = aws_get_retry_time(), 
network_timeout = aws_get_network_timeout(), 
region = aws_get_region()
)
```

Arguments

- **VpcId**: Character. The ID of the VPC in which the endpoint will be used.
- **ServiceName**: Character. The service name.
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
- **VpcEndpointType**: Character. The type of endpoint. Default: Gateway [optional]
- **PolicyDocument**: Character. (Interface and gateway endpoints) A policy to attach to the endpoint that controls access to the...[optional]
- **RouteTableId**: List. (Gateway endpoint) One or more route table IDs.[optional]
SubnetId
List. (Interface and Gateway Load Balancer endpoints) The ID of one or more
subnets in which to create...[optional]

SecurityGroupId
List. (Interface endpoint) The ID of one or more security groups to associate
with the endpoint network...[optional]

ClientToken
Character. Unique, case-sensitive identifier that you provide to ensure the idem-
potency of the request.[optional]

PrivateDnsEnabled
Logical. (Interface endpoint) Indicates whether to associate a private hosted
zone with the specified VPC.[optional]

TagSpecification
List. The tags to associate with the endpoint.[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and
need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network
issue. If the request has been sent retry_time times but still not be able to get
the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can
not be less than 1 ms.

region
Character. The region of the AWS service.

Value
A list object or a character vector

VpcId
The ID of the VPC in which the endpoint will be used.

ServiceName
The service name. To get a list of available services, use the DescribeVpcEndpointServices request,
or get the name from the service provider.

DryRun
Checks whether you have the required permissions for the action, without actually making the
request, and provides an error response. If you have the required permissions, the error response is
DryRunOperation. Otherwise, it is UnauthorizedOperation.

VpcEndpointType
The type of endpoint.
Default: Gateway
PolicyDocument

(Interface and gateway endpoints) A policy to attach to the endpoint that controls access to the service. The policy must be in valid JSON format. If this parameter is not specified, we attach a default policy that allows full access to the service.

RouteTableId

(Gateway endpoint) One or more route table IDs.

SubnetId

(Interface and Gateway Load Balancer endpoints) The ID of one or more subnets in which to create an endpoint network interface. For a Gateway Load Balancer endpoint, you can specify one subnet only.

SecurityGroupId

(Interface endpoint) The ID of one or more security groups to associate with the endpoint network interface.

ClientToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. For more information, see How to Ensure Idempotency.

PrivateDnsEnabled

(Interface endpoint) Indicates whether to associate a private hosted zone with the specified VPC. The private hosted zone contains a record set for the default public DNS name for the service for the Region (for example, kinesis.us-east-1.amazonaws.com), which resolves to the private IP addresses of the endpoint network interfaces in the VPC. This enables you to make requests to the default public DNS name for the service instead of the public DNS names that are automatically generated by the VPC endpoint service.

To use a private hosted zone, you must set the following VPC attributes to true: enableDnsHostnames and enableDnsSupport. Use ModifyVpcAttribute to set the VPC attributes.

Default: true

TagSpecification

The tags to associate with the endpoint.
Create Vpc Endpoint Connection Notification

**Description**
Create Vpc Endpoint Connection Notification

**Usage**

```r
ec2_create_vpc_endpoint_connection_notification(
    ConnectionNotificationArn,
    ConnectionEvents,
    DryRun = NULL,
    ServiceId = NULL,
    VpcEndpointId = NULL,
    ClientToken = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

**Arguments**

- **ConnectionNotificationArn**
  Character. The ARN of the SNS topic for the notifications.

- **ConnectionEvents**
  List. One or more endpoint events for which to receive notifications.

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

- **ServiceId**
  Character. The ID of the endpoint service.[optional]

- **VpcEndpointId**
  Character. The ID of the endpoint.[optional]

- **ClientToken**
  Character. Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.[optional]

- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response.[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout
    Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
    Character. The region of the AWS service.

Value
A list object or a character vector

ConnectionNotificationArn
The ARN of the SNS topic for the notifications.

ConnectionEvents
One or more endpoint events for which to receive notifications. Valid values are Accept, Connect, Delete, and Reject.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ServiceId
The ID of the endpoint service.

VpcEndpointId
The ID of the endpoint.

ClientToken
Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. For more information, see How to Ensure Idempotency.

Description
Create Vpc Endpoint Service Configuration
Usage

e2_create_vpc_endpoint_service_configuration(
    DryRun = NULL,
    AcceptanceRequired = NULL,
    PrivateDnsName = NULL,
    NetworkLoadBalancerArn = NULL,
    GatewayLoadBalancerArn = NULL,
    ClientToken = NULL,
    TagSpecification = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

AcceptanceRequired Logical. Indicates whether requests from service consumers to create an endpoint to your service must be accepted.[optional]

PrivateDnsName Character. (Interface endpoint configuration) The private DNS name to assign to the VPC endpoint service.[optional]

NetworkLoadBalancerArn List. The Amazon Resource Names (ARNs) of one or more Network Load Balancers for your service.[optional]

GatewayLoadBalancerArn List. The Amazon Resource Names (ARNs) of one or more Gateway Load Balancers.[optional]

ClientToken Character. Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.[optional]

TagSpecification List. The tags to associate with the service.[optional]

simplify Logical. Whether to simplify the result and handle nextToken in the response.[optional]

others Named list. The parameters that are not included in the function parameters and need to be added into the request.[optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.
Value

A list object or a character vector

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

AcceptanceRequired

Indicates whether requests from service consumers to create an endpoint to your service must be accepted. To accept a request, use AcceptVpcEndpointConnections.

PrivateDnsName

(Interface endpoint configuration) The private DNS name to assign to the VPC endpoint service.

NetworkLoadBalancerArn

The Amazon Resource Names (ARNs) of one or more Network Load Balancers for your service.

GatewayLoadBalancerArn

The Amazon Resource Names (ARNs) of one or more Gateway Load Balancers.

ClientToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. For more information, see How to Ensure Idempotency.

TagSpecification

The tags to associate with the service.
Usage

ec2_create_vpc_peering_connection(
    DryRun = NULL,
    PeerOwnerId = NULL,
    PeerVpcId = NULL,
    VpcId = NULL,
    PeerRegion = NULL,
    TagSpecification = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

PeerOwnerId Character. The AWS account ID of the owner of the accepter VPC. Default: Your AWS account ID [optional]

PeerVpcId Character. The ID of the VPC with which you are creating the VPC peering connection.[optional]

VpcId Character. The ID of the requester VPC. You must specify this parameter in the request.[optional]

PeerRegion Character. The Region code for the accepter VPC, if the accepter VPC is located in a Region other than the...[optional]

TagSpecification List. The tags to assign to the peering connection.[optional]

simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]

others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.

Value

A list object or a character vector
**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

**PeerOwnerId**

The AWS account ID of the owner of the accepter VPC.

Default: Your AWS account ID

**PeerVpcId**

The ID of the VPC with which you are creating the VPC peering connection. You must specify this parameter in the request.

**VpcId**

The ID of the requester VPC. You must specify this parameter in the request.

**PeerRegion**

The Region code for the accepter VPC, if the accepter VPC is located in a Region other than the Region in which you make the request.

Default: The Region in which you make the request.

**TagSpecification**

The tags to assign to the peering connection.

---

```
ec2_create_vpn_connection

Create Vpn Connection
```

---

**Description**

Create Vpn Connection

**Usage**

```python
ec2_create_vpn_connection(
    CustomerGatewayId,
    Type,
    VpnGatewayId = NULL,
    TransitGatewayId = NULL,
    DryRun = NULL,
    Options = NULL,
    TagSpecification = NULL,
```
Arguments

CustomerGatewayId
   Character. The ID of the customer gateway.

Type
   Character. The type of VPN connection (ipsec.1).

VpnGatewayId
   Character. The ID of the virtual private gateway.[optional]

TransitGatewayId
   Character. The ID of the transit gateway.[optional]

DryRun
   Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]

Options
   Object. The options for the VPN connection.[optional]

TagSpecification
   List. The tags to apply to the VPN connection.[optional]

simplify
   Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
   Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
   Logical. Whether to show an error message when a network error occurs.

retry_time
   Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
   Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
   Character. The region of the AWS service.

Value

A list object or a character vector

CustomerGatewayId

The ID of the customer gateway.

Type

The type of VPN connection (ipsec.1).
VpnGatewayId

The ID of the virtual private gateway. If you specify a virtual private gateway, you cannot specify a transit gateway.

TransitGatewayId

The ID of the transit gateway. If you specify a transit gateway, you cannot specify a virtual private gateway.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Options

The options for the VPN connection.

TagSpecification

The tags to apply to the VPN connection.

---

**ec2_create_vpn_connection_route**

*Create Vpn Connection Route*

**Description**

Create Vpn Connection Route

**Usage**

```r
ec2_create_vpn_connection_route(
  DestinationCidrBlock,
  VpnConnectionId,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
Arguments

- **DestinationCidrBlock**: Character. The CIDR block associated with the local subnet of the customer network.
- **VpnConnectionId**: Character. The ID of the VPN connection.
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response [optional].
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request [optional].
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

**DestinationCidrBlock**

The CIDR block associated with the local subnet of the customer network.

**VpnConnectionId**

The ID of the VPN connection.

---

**ec2_create_vpn_gateway**

*Create Vpn Gateway*

---

Description

Create Vpn Gateway

Usage

```r
ec2_create_vpn_gateway(
  Type,
  AvailabilityZone = NULL,
  TagSpecification = NULL,
  AmazonSideAsn = NULL,
)```
DryRun = NULL,
simplify = TRUE,
others = list(),
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

<table>
<thead>
<tr>
<th>Type</th>
<th>Character. The type of VPN connection this virtual private gateway supports.</th>
</tr>
</thead>
<tbody>
<tr>
<td>AvailabilityZone</td>
<td>Character. The Availability Zone for the virtual private gateway.[optional]</td>
</tr>
<tr>
<td>TagSpecification</td>
<td>List. The tags to apply to the virtual private gateway.[optional]</td>
</tr>
<tr>
<td>AmazonSideAsn</td>
<td>Integer. A private Autonomous System Number (ASN) for the Amazon side of a BGP session.[optional]</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response[optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>

Value

A list object or a character vector

Type

The type of VPN connection this virtual private gateway supports.

AvailabilityZone

The Availability Zone for the virtual private gateway.

TagSpecification

The tags to apply to the virtual private gateway.
AmazonSideAsn
A private Autonomous System Number (ASN) for the Amazon side of a BGP session. If you're using a 16-bit ASN, it must be in the 64512 to 65534 range. If you're using a 32-bit ASN, it must be in the 4200000000 to 4294967294 range.
Default: 64512

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Usage
ec2_delete_carrier_gateway(  CarrierGatewayId,  DryRun = NULL,  simplify = TRUE,  others = list(),  print_on_error = aws_get_print_on_error(),  retry_time = aws_get_retry_time(),  network_timeout = aws_get_network_timeout(),  region = aws_get_region() )

Arguments
CarrierGatewayId
Character. The ID of the carrier gateway.

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
\textbf{network\_timeout}

Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

\textbf{region}

Character. The region of the AWS service.

\textbf{Value}

A list object or a character vector

\textbf{Carrier\textunderscore GatewayId}

The ID of the carrier gateway.

\textbf{DryRun}

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is \texttt{DryRun\textunderscore Operation}. Otherwise, it is \texttt{Unauthorized\textunderscore Operation}.

---

\texttt{ec2\_delete\_client\_vpn\_endpoint}

\textit{Delete Client Vpn Endpoint}

\textbf{Description}

Deletes the specified Client VPN endpoint. You must disassociate all target networks before you can delete a Client VPN endpoint.

\textbf{Usage}

\begin{verbatim}
ec2_delete_client_vpn_endpoint(
    ClientVpnEndpointId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
\end{verbatim}

\textbf{Arguments}

\begin{itemize}
  \item \texttt{ClientVpnEndpointId}  
    Character. The ID of the Client VPN to be deleted.
  \item \texttt{DryRun}  
    Logical. Checks whether you have the required permissions for the action, without actually making the request. ... [optional]
\end{itemize}
ec2_delete_client_vpn_route

simplify Logical. Whether to simplify the result and handle nextToken in the response [optional]

others Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

deregion Character. The region of the AWS service.

Value
A list object or a character vector

ClientVpnEndpointId
The ID of the Client VPN to be deleted.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Usage

c2.delete_client_vpn_route(
  ClientVpnEndpointId,
  DestinationCidrBlock,
  TargetVpcSubnetId = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),

Description
Deletes a route from a Client VPN endpoint. You can only delete routes that you manually added using the CreateClientVpnRoute action. You cannot delete routes that were automatically added when associating a subnet. To remove routes that have been automatically added, disassociate the target subnet from the Client VPN endpoint.
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

ClientVpnEndpointId
Character. The ID of the Client VPN endpoint from which the route is to be deleted.

DestinationCidrBlock
Character. The IPv4 address range, in CIDR notation, of the route to be deleted.

TargetVpcSubnetId
Character. The ID of the target subnet used by the route.[optional]

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]

simplify
Logical. Whether to simplify the result and handle nextState in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value

A list object or a character vector

ClientVpnEndpointId
The ID of the Client VPN endpoint from which the route is to be deleted.

DestinationCidrBlock
The IPv4 address range, in CIDR notation, of the route to be deleted.

TargetVpcSubnetId
The ID of the target subnet used by the route.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_delete_customer_gateway
Delete Customer Gateway

Description

Deletes the specified customer gateway. You must delete the VPN connection before you can delete the customer gateway.

Usage

c2_delete_customer_gateway(
    CustomerGatewayId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

CustomerGatewayId
Character. The ID of the customer gateway.

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.
Value

A list object or a character vector

CustomerGatewayId

The ID of the customer gateway.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**ec2_delete_dhcp_options**

*Delete Dhcp Options*

---

**Description**

Deletes the specified set of DHCP options. You must disassociate the set of DHCP options before you can delete it. You can disassociate the set of DHCP options by associating either a new set of options or the default set of options with the VPC.

**Usage**

```r
ec2_delete_dhcp_options(
  DhcpOptionsId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DhcpOptionsId</td>
<td>Character. The ID of the DHCP options set.</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response. [optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
</tbody>
</table>
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  Character. The region of the AWS service.

Value
A list object or a character vector

DhcpOptionsId
The ID of the DHCP options set.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Usage
```r
ec2_delete_egress_only_internet_gateway(
  EgressOnlyInternetGatewayId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
ec2_delete_fleets

Arguments

EgressOnlyInternetGatewayId
Character. The ID of the egress-only internet gateway.

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value
A list object or a character vector

EgressOnlyInternetGatewayId
The ID of the egress-only internet gateway.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

Description
Delete Fleets

Usage

ec2_delete_fleets(
    FleetId,
    TerminateInstances,
    DryRun = NULL,
    simplify = TRUE,
)
ec2_delete_fleets

```python
others = list(),
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FleetId</td>
<td>List. The IDs of the EC2 Fleets.</td>
</tr>
<tr>
<td>TerminateInstances</td>
<td>Logical. Indicates whether to terminate the instances when the EC2 Fleet is deleted.</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response[optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>

**Value**

A list object or a character vector

**FleetId**

The IDs of the EC2 Fleets.

**TerminateInstances**

Indicates whether to terminate the instances when the EC2 Fleet is deleted. The default is to terminate the instances.

To let the instances continue to run after the EC2 Fleet is deleted, specify NoTerminateInstances. Supported only for fleets of type maintain and request.

For instant fleets, you cannot specify NoTerminateInstances. A deleted instant fleet with running instances is not supported.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

**ec2_delete_flow_logs**  Delete Flow Logs

**Description**

Deletes one or more flow logs.

**Usage**

```r
ec2_delete_flow_logs(
  FlowLogId,  # List. One or more flow log IDs. Constraint: Maximum of 1000 flow log IDs.
  DryRun = NULL,  # Logical. Checks whether you have the required permissions for the action, without actually making the request, ...[optional]
  simplify = TRUE,  # Logical. Whether to simplify the result and handle nextToken in the response[optional]
  others = list(),  # Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
  print_on_error = aws_get_print_on_error(),  # Logical. Whether to show an error message when a network error occurs.
  retry_time = aws_get_retry_time(),  # Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
  network_timeout = aws_get_network_timeout(),  # Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
  region = aws_get_region()  # Character. The region of the AWS service.
)
```

**Arguments**

- **FlowLogId**: List. One or more flow log IDs. Constraint: Maximum of 1000 flow log IDs.
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request, ...[optional]
- **simplify**: Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

**Value**

A list object or a character vector
FlowLogId

One or more flow log IDs.
Constraint: Maximum of 1000 flow log IDs.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

description

Delete the specified Amazon FPGA Image (AFI).

Usage

dc2_delete_fpga_image(
    FpgaImageId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

FpgaImageId  Character. The ID of the AFI.
DryRun       Logical. Checks whether you have the required permissions for the action, without actually making the request, [optional]
simplify    Logical. Whether to simplify the result and handle nextToken in the response[optional]
others      Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region      Character. The region of the AWS service.
Value

A list object or a character vector

**FpgaImageId**

The ID of the AFI.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**ec2_delete_internet_gateway**

*Delete Internet Gateway*

---

**Description**

Deletes the specified internet gateway. You must detach the internet gateway from the VPC before you can delete it.

**Usage**

```r
ec2_delete_internet_gateway(
    InternetGatewayId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

**Arguments**

- **InternetGatewayId**
  Character. The ID of the internet gateway.

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]

- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.
**retry_time**  
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

**network_timeout**  
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

**region**  
Character. The region of the AWS service.

**Value**

A list object or a character vector

**InternetGatewayId**

The ID of the internet gateway.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

---

**ec2_delete_key_pair**  
**Delete Key Pair**

**Description**

Deletes the specified key pair, by removing the public key from Amazon EC2.

**Usage**

```r
ec2_delete_key_pair(
  KeyName = NULL,
  KeyPairId = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
Arguments

KeyName Character. The name of the key pair.[optional]
KeyPairId Character. The ID of the key pair.[optional]
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request,[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value

A list object or a character vector

KeyName

The name of the key pair.

KeyPairId

The ID of the key pair.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description

Deletes a launch template. Deleting a launch template deletes all of its versions.
Usage

ec2_delete_launch_template(
  DryRun = NULL,
  LaunchTemplateId = NULL,
  LaunchTemplateName = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
LaunchTemplateId Character. The ID of the launch template.[optional]
LaunchTemplateName Character. The name of the launch template.[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value

A list object or a character vector

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

LaunchTemplateId

The ID of the launch template. You must specify either the launch template ID or launch template name in the request.
LaunchTemplateName

The name of the launch template. You must specify either the launch template ID or launch template name in the request.

---

### Description

Deletes one or more versions of a launch template. You cannot delete the default version of a launch template; you must first assign a different version as the default. If the default version is the only version for the launch template, you must delete the entire launch template using DeleteLaunchTemplate.

### Usage

```r
ec2_delete_launch_template_versions(
  LaunchTemplateVersion,
  DryRun = NULL,
  LaunchTemplateId = NULL,
  LaunchTemplateName = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

### Arguments

- **LaunchTemplateVersion**
  - List. The version numbers of one or more launch template versions to delete.

- **DryRun**
  - Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

- **LaunchTemplateId**
  - Character. The ID of the launch template.[optional]

- **LaunchTemplateName**
  - Character. The name of the launch template.[optional]

- **simplify**
  - Logical. Whether to simplify the result and handle nextToken in the response[optional]

- **others**
  - Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  - Logical. Whether to show an error message when a network error occurs.
**retry_time**  
Integer. Number of retries for a REST request when encountering the network issue. If the request has been sent `retry_time` times but still not able to get the response, an error will be thrown.

**network_timeout**  
Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.

**region**  
Character. The region of the AWS service.

### Value

A list object or a character vector.

### LaunchTemplateVersion

The version numbers of one or more launch template versions to delete.

### DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

### LaunchTemplateId

The ID of the launch template. You must specify either the launch template ID or launch template name in the request.

### LaunchTemplateName

The name of the launch template. You must specify either the launch template ID or launch template name in the request.

---

**ec2_delete_local_gateway_route**

*Delete Local Gateway Route*

---

**Description**

Deletes the specified route from the specified local gateway route table.

**Usage**

```r
ec2_delete_local_gateway_route(
  DestinationCidrBlock,
  LocalGatewayRouteTableId,
  DryRun = NULL,
  simplify = TRUE,
)```
others = list(),
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

**Arguments**

**DestinationCidrBlock**
Character. The CIDR range for the route. This must match the CIDR for the route exactly.

**LocalGatewayRouteTableId**
Character. The ID of the local gateway route table.

**DryRun**
Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

**simplify**
Logical. Whether to simplify the result and handle `nextToken` in the response[optional]

**others**
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

**print_on_error**
Logical. Whether to show an error message when a network error occurs.

**retry_time**
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

**network_timeout**
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

**region**
Character. The region of the AWS service.

**Value**
A list object or a character vector

**DestinationCidrBlock**
The CIDR range for the route. This must match the CIDR for the route exactly.

**LocalGatewayRouteTableId**
The ID of the local gateway route table.

**DryRun**
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`. 
Delete Local Gateway Route Table Vpc Association

**Description**

Deletes the specified association between a VPC and local gateway route table.

**Usage**

```r
ec2_delete_local_gateway_route_table_vpc_association(
    LocalGatewayRouteTableVpcAssociationId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

**Arguments**

- **LocalGatewayRouteTableVpcAssociationId**
  Character. The ID of the association.
- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]
- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]
- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.
- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**
  Character. The region of the AWS service.

**Value**

A list object or a character vector

**LocalGatewayRouteTableVpcAssociationId**

The ID of the association.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description

Deletes the specified managed prefix list. You must first remove all references to the prefix list in your resources.

Usage

```r
ec2_delete_managed_prefix_list(
  PrefixListId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **PrefixListId** Character. The ID of the prefix list.
- **DryRun** Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **simplify** Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]
- **others** Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]
- **print_on_error** Logical. Whether to show an error message when a network error occurs.
- **retry_time** Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout** Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region** Character. The region of the AWS service.
ec2_delete_nat_gateway

Value

A list object or a character vector

PrefixListId

The ID of the prefix list.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_delete_nat_gateway

Delete Nat Gateway

Description

Deletes the specified NAT gateway. Deleting a NAT gateway disassociates its Elastic IP address, but does not release the address from your account. Deleting a NAT gateway does not delete any NAT gateway routes in your route tables.

Usage

```r
ec2_delete_nat_gateway(
  NatGatewayId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **NatGatewayId**  Character. The ID of the NAT gateway.
- **DryRun** Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **simplify** Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]
- **others** Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]
- **print_on_error** Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  Character. The region of the AWS service.

Value
A list object or a character vector

NatGatewayId
The ID of the NAT gateway.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Usage:
```r
e2_delete_network_acl(
  NetworkAclId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)```

Description
Deletes the specified network ACL. You can't delete the ACL if it's associated with any subnets. You can't delete the default network ACL.
Arguments

NetworkAclId  Character. The ID of the network ACL.
DryRun        Logical. Checks whether you have the required permissions for the action, without actually making the request, [optional]
simplify      Logical. Whether to simplify the result and handle nextToken in the response [optional]
others        Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time    Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region        Character. The region of the AWS service.

Value

A list object or a character vector

NetworkAclId

The ID of the network ACL.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description

Deletes the specified ingress or egress entry (rule) from the specified network ACL.
Usage

```r
ec2_delete_network_acl_entry(
    Egress,
    NetworkAclId,
    RuleNumber,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **Egress**  
  Logical. Indicates whether the rule is an egress rule.
- **NetworkAclId**  
  Character. The ID of the network ACL.
- **RuleNumber**  
  Integer. The rule number of the entry to delete.
- **DryRun**  
  Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **simplify**  
  Logical. Whether to simplify the result and handle nextToken in the response [optional]
- **others**  
  Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]
- **print_on_error**  
  Logical. Whether to show an error message when a network error occurs.
- **retry_time**  
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- **network_timeout**  
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**  
  Character. The region of the AWS service.

Value

A list object or a character vector

Egress

Indicates whether the rule is an egress rule.

NetworkAclId

The ID of the network ACL.
**RuleNumber**

The rule number of the entry to delete.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**ec2_delete_network_insights_analysis**

*Delete Network Insights Analysis*

---

**Description**

Deletes the specified network insights analysis.

**Usage**

```r
ec2_delete_network_insights_analysis(
    NetworkInsightsAnalysisId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

**Arguments**

- **NetworkInsightsAnalysisId**
  Character. The ID of the network insights analysis.

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]

- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  Character. The region of the AWS service.
Value

A list object or a character vector

NetworkInsightsAnalysisId

The ID of the network insights analysis.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description

Deletes the specified path.

Usage

```r
e2_delete_network_insights_path(
  NetworkInsightsPathId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **NetworkInsightsPathId**
  Character. The ID of the path.

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request…[optional]

- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.
### ec2_delete_network_interface

#### RetryTime

Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

#### NetworkTimeout

Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.

#### Region

Character. The region of the AWS service.

### Value

A list object or a character vector

### NetworkInsightsPathId

The ID of the path.

### DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

```r
ec2_delete_network_interface

Delete Network Interface

Description

Deletes the specified network interface. You must detach the network interface before you can delete it.

Usage

ec2_delete_network_interface(
    NetworkInterfaceId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)`
Arguments

**NetworkInterfaceId**
Character. The ID of the network interface.

**DryRun**
Logical. Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

**simplify**
Logical. Whether to simplify the result and handle nextToken in the response.

**others**
Named list. The parameters that are not included in the function parameters and need to be added into the request.

**print_on_error**
Logical. Whether to show an error message when a network error occurs.

**retry_time**
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

**network_timeout**
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

**region**
Character. The region of the AWS service.

Value

A list object or a character vector

**NetworkInterfaceId**
The ID of the network interface.

**DryRun**
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description

Deletes a permission for a network interface. By default, you cannot delete the permission if the account for which you're removing the permission has attached the network interface to an instance. However, you can force delete the permission, regardless of any attachment.
Usage

```r
ec2_delete_network_interface_permission(
    NetworkInterfacePermissionId,
    Force = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **NetworkInterfacePermissionId**
  Character. The ID of the network interface permission.

- **Force**
  Logical. Specify `true` to remove the permission even if the network interface is attached to an instance.[optional]

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]

- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  Character. The region of the AWS service.

Value

A list object or a character vector

**NetworkInterfacePermissionId**

The ID of the network interface permission.

**Force**

Specify `true` to remove the permission even if the network interface is attached to an instance.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Usage

e2.delete_placement_group(
  GroupName,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws.get_print_on_error(),
  retry_time = aws.get_retry_time(),
  network_timeout = aws.get_network_timeout(),
  region = aws.get_region()
)

Arguments

GroupName     Character. The name of the placement group.
DryRun        Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]
simplify      Logical. Whether to simplify the result and handle nextToken in the response[optional]
others        Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time    Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region        Character. The region of the AWS service.
**Value**
A list object or a character vector

**GroupName**
The name of the placement group.

**DryRun**
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**ec2_delete_queued_reserved_instances**

*Delete Queued Reserved Instances*

---

**Description**
Deletes the queued purchases for the specified Reserved Instances.

**Usage**

```r
ec2_delete_queued_reserved_instances(
  ReservedInstancesId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- **ReservedInstancesId**
  List. The IDs of the Reserved Instances.
- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]
- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]
- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.
### Description

Deletes the specified route from the specified route table.

### Usage

```r
ec2_delete_route(
  RouteTableId,
  DestinationCidrBlock = NULL,
  DestinationIpv6CidrBlock = NULL,
  DestinationPrefixListId = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
Arguments

- **RouteTableId**  
  Character. The ID of the route table.

- **DestinationCidrBlock**  
  Character. The IPv4 CIDR range for the route. The value you specify must match the CIDR for the route exactly. [optional]

- **DestinationIpv6CidrBlock**  
  Character. The IPv6 CIDR range for the route. The value you specify must match the CIDR for the route exactly. [optional]

- **DestinationPrefixListId**  
  Character. The ID of the prefix list for the route. [optional]

- **DryRun**  
  Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]

- **simplify**  
  Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]

- **others**  
  Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]

- **print_on_error**  
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**  
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

- **network_timeout**  
  Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.

- **region**  
  Character. The region of the AWS service.

Value

A list object or a character vector

- **RouteTableId**  
  The ID of the route table.

- **DestinationCidrBlock**  
  The IPv4 CIDR range for the route. The value you specify must match the CIDR for the route exactly.

- **DestinationIpv6CidrBlock**  
  The IPv6 CIDR range for the route. The value you specify must match the CIDR for the route exactly.

- **DestinationPrefixListId**  
  The ID of the prefix list for the route.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

description

Deletes the specified route table. You must disassociate the route table from any subnets before you can delete it. You can’t delete the main route table.

ec2_delete_route_table

Delete Route Table

Usage

def ec2_delete_route_table(
    RouteTableId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

RouteTableId Character. The ID of the route table.
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.
Value

A list object or a character vector

RouteTableId

The ID of the route table.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description

Delete Security Group

Usage

```r
df2_delete_security_group(
    GroupId = NULL,
    GroupName = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GroupId</td>
<td>Character. The ID of the security group. Required for a nondefault VPC.</td>
</tr>
<tr>
<td>GroupName</td>
<td>Character. [EC2-Classic, default VPC] The name of the security group.</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request.</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response.</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request.</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
</tbody>
</table>
### retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

### network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

### region
Character. The region of the AWS service.

#### Value
A list object or a character vector

#### GroupId
The ID of the security group. Required for a nondefault VPC.

#### GroupName
[EC2-Classic, default VPC] The name of the security group. You can specify either the security group name or the security group ID.

#### DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

### ec2_delete_snapshot  Delete Snapshot

#### Description
Delete Snapshot

#### Usage
```r
ec2_delete_snapshot(
  SnapshotId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
Arguments

- **SnapshotId**: Character. The ID of the EBS snapshot.
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

**SnapshotId**

The ID of the EBS snapshot.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is **DryRunOperation**. Otherwise, it is **UnauthorizedOperation**.

Description

Deletes the data feed for Spot Instances.
Usage

```r
c2_delete_subnet_datafeed_subscription(
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **DryRun** (Logical): Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **simplify** (Logical): Whether to simplify the result and handle `nextToken` in the response. [optional]
- **others** (Named list): The parameters that are not included in the function parameters and need to be added into the request. [optional]
- **print_on_error** (Logical): Whether to show an error message when a network error occurs.
- **retry_time** (Integer): Number of retries for a REST request when encountering a network issue. If the request has been sent `retry_time` times but still not able to get the response, an error will be thrown.
- **network_timeout** (Numeric): Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.
- **region** (Character): The region of the AWS service.

Value

A list object or a character vector

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

Description

Deletes the specified subnet. You must terminate all running instances in the subnet before you can delete the subnet.
Usage

```r
ec2_delete_subnet(
    SubnetId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **SubnetId**  
  Character. The ID of the subnet.

- **DryRun**  
  Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

- **simplify**  
  Logical. Whether to simplify the result and handle `nextToken` in the response[optional]

- **others**  
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**  
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**  
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

- **network_timeout**  
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**  
  Character. The region of the AWS service.

Value

A list object or a character vector

**SubnetId**

The ID of the subnet.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`. 
ec2_delete_tags

Delete Tags

Description
Delete Tags

Usage
ec2_delete_tags(
    ResourceId,
    DryRun = NULL,
    Tag = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments
ResourceId      List. The IDs of the resources, separated by spaces.
DryRun          Logical. Checks whether you have the required permissions for the action, without actually making the request,[optional]
Tag             List. The tags to delete.[optional]
simplify        Logical. Whether to simplify the result and handle nextToken in the response[optional]
others           Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error   Logical. Whether to show an error message when a network error occurs.
retry_time       Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region          Character. The region of the AWS service.

Value
A list object or a character vector
**ResourceId**

The IDs of the resources, separated by spaces. Constraints: Up to 1000 resource IDs. We recommend breaking up this request into smaller batches.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is **DryRunOperation**. Otherwise, it is **UnauthorizedOperation**.

**Tag**

The tags to delete. Specify a tag key and an optional tag value to delete specific tags. If you specify a tag key without a tag value, we delete any tag with this key regardless of its value. If you specify a tag key with an empty string as the tag value, we delete the tag only if its value is an empty string. If you omit this parameter, we delete all user-defined tags for the specified resources. We do not delete AWS-generated tags (tags that have the **aws:** prefix).

---

**ec2_delete_traffic_mirror_filter**

*Delete Traffic Mirror Filter*

---

**Description**

Delete Traffic Mirror Filter

**Usage**

```r
ec2_delete_traffic_mirror_filter(
  TrafficMirrorFilterId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- **TrafficMirrorFilterId**: Character. The ID of the Traffic Mirror filter.
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]
Others

Named list. The parameters that are not included in the function parameters and need to be added into the request [optional].

Print on error

Logical. Whether to show an error message when a network error occurs.

Retry time

Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

Network timeout

Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

Region

Character. The region of the AWS service.

Value

A list object or a character vector

TrafficMirrorFilterId

The ID of the Traffic Mirror filter.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description

Deletes the specified Traffic Mirror rule.

Usage

```
ec2_delete_traffic_mirror_filter_rule(
    TrafficMirrorFilterRuleId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```
Arguments

TrafficMirrorFilterRuleId
Character. The ID of the Traffic Mirror rule.

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value

A list object or a character vector

TrafficMirrorFilterRuleId

The ID of the Traffic Mirror rule.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description

Deletes the specified Traffic Mirror session.
Usage

```r
e2_delete_traffic_mirror_session(
    TrafficMirrorSessionId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **TrafficMirrorSessionId**
  Character. The ID of the Traffic Mirror session.

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  Character. The region of the AWS service.

Value

A list object or a character vector

**TrafficMirrorSessionId**

The ID of the Traffic Mirror session.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.
Delete Traffic Mirror Target

Usage

```r
ec2_delete_traffic_mirror_target(
    TrafficMirrorTargetId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **TrafficMirrorTargetId**: Character. The ID of the Traffic Mirror target.
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

TrafficMirrorTargetId

The ID of the Traffic Mirror target.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

```
ec2_delete_transit_gateway

Delete Transit Gateway
```

Description

Deletes the specified transit gateway.

Usage

```
ec2_delete_transit_gateway(
    TransitGatewayId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

TransitGatewayId

Character. The ID of the transit gateway.

DryRun

Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

simplify

Logical. Whether to simplify the result and handle nextToken in the response[optional]

others

Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error

Logical. Whether to show an error message when a network error occurs.

retry_time

Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout

Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region

Character. The region of the AWS service.
Value
A list object or a character vector

TransitGatewayId
The ID of the transit gateway.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description
Deletes the specified Connect attachment. You must first delete any Connect peers for the attachment.

Usage
```r
ec2_delete_transit_gateway_connect(
   TransitGatewayAttachmentId,
   DryRun = NULL,
   simplify = TRUE,
   others = list(),
   print_on_error = aws_get_print_on_error(),
   retry_time = aws_get_retry_time(),
   network_timeout = aws_get_network_timeout(),
   region = aws_get_region()
)
```

Arguments

TransitGatewayAttachmentId
Character. The ID of the Connect attachment.

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  Character. The region of the AWS service.

Value
A list object or a character vector

TransitGatewayAttachmentId
The ID of the Connect attachment.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Usage

```r
ec2_delete_transit_gateway_connect_peer(
  TransitGatewayConnectPeerId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
### ec2_delete_transit_gateway_multicast_domain

**Delete Transit Gateway Multicast Domain**

**Description**

Deletes the specified transit gateway multicast domain.

**Arguments**

- **TransitGatewayConnectPeerId**
  Character. The ID of the Connect peer.

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request, [...][optional]

- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  Character. The region of the AWS service.

**Value**

A list object or a character vector

**TransitGatewayConnectPeerId**

The ID of the Connect peer.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`. 
Usage

```
ec2_delete_transit_gateway_multicast_domain(
    TransitGatewayMulticastDomainId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **TransitGatewayMulticastDomainId**: Character. The ID of the transit gateway multicast domain.
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response [optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encountering the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

**TransitGatewayMulticastDomainId**

The ID of the transit gateway multicast domain.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`. 
Description

Deletes a transit gateway peering attachment.

Usage

ec2_delete_transit_gateway_peering_attachment(
    TransitGatewayAttachmentId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

TransitGatewayAttachmentId
    Character. The ID of the transit gateway peering attachment.

DryRun
    Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]

simplify
    Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
    Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
    Logical. Whether to show an error message when a network error occurs.

retry_time
    Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
    Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
    Character. The region of the AWS service.

Value

A list object or a character vector

TransitGatewayAttachmentId

The ID of the transit gateway peering attachment.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

Delete Transit Gateway Prefix List Reference

Deletes a reference (route) to a prefix list in a specified transit gateway route table.

Usage

```r
ec2_delete_transit_gateway_prefix_list_reference(
  TransitGatewayRouteTableId,
  PrefixListId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **TransitGatewayRouteTableId**  
  Character. The ID of the route table.
- **PrefixListId**  
  Character. The ID of the prefix list.
- **DryRun**  
  Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
- **simplify**  
  Logical. Whether to simplify the result and handle nextToken in the response.[optional]
- **others**  
  Named list. The parameters that are not included in the function parameters and need to be added into the request.[optional]
- **print_on_error**  
  Logical. Whether to show an error message when a network error occurs.
- **retry_time**  
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- **network_timeout**  
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**  
  Character. The region of the AWS service.
**Value**

A list object or a character vector

**TransitGatewayRouteTableId**

The ID of the route table.

**PrefixListId**

The ID of the prefix list.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

---

---

**Description**

Delete the specified route from the specified transit gateway route table.

**Usage**

```r
ec2_delete_transit_gateway_route(
  TransitGatewayRouteTableId,
  DestinationCidrBlock,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

**TransitGatewayRouteTableId**

Character. The ID of the transit gateway route table.

**DestinationCidrBlock**

Character. The CIDR range for the route. This must match the CIDR for the route exactly.
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]

simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]

others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
	network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.

Value

A list object or a character vector

TransitGatewayRouteTableId

The ID of the transit gateway route table.

DestinationCidrBlock

The CIDR range for the route. This must match the CIDR for the route exactly.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description

Deletes the specified transit gateway route table. You must disassociate the route table from any transit gateway route tables before you can delete it.
Usage

```r
e2_delete_transit_gateway_route_table(
  TransitGatewayRouteTableId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **TransitGatewayRouteTableId**: Character. The ID of the transit gateway route table.
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request,[optional]
- **simplify**: Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

**TransitGatewayRouteTableId**

The ID of the transit gateway route table.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
Delete Transit Gateway Vpc Attachment

Description

Deletes the specified VPC attachment.

Usage

ec2_delete_transit_gateway_vpc_attachment(
  TransitGatewayAttachmentId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

TransitGatewayAttachmentId
  Character. The ID of the attachment.

DryRun
  Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

simplify
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
  Logical. Whether to show an error message when a network error occurs.

retry_time
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
  Character. The region of the AWS service.

Value

A list object or a character vector

TransitGatewayAttachmentId

The ID of the attachment.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

Description

Delete Volume

Usage

c2_delete_volume(
  VolumeId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VolumeId</td>
<td>Character. The ID of the volume.</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request,[optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response[optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>

Value

A list object or a character vector
VolumeId

The ID of the volume.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**ec2_delete_vpc**

**Delete Vpc**

---

**Description**

Deletes the specified VPC. You must detach or delete all gateways and resources that are associated with the VPC before you can delete it. For example, you must terminate all instances running in the VPC, delete all security groups associated with the VPC (except the default one), delete all route tables associated with the VPC (except the default one), and so on.

**Usage**

ec2_delete_vpc(
  VpcId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

**Arguments**

- **VpcId**
  Character. The ID of the VPC.
- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response.[optional]
- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request.[optional]
- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.
- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
network_timeout
 Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
 Character. The region of the AWS service.

Value
 A list object or a character vector

VpcId
 The ID of the VPC.

DryRun
 Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_delete_vpc_endpoints
 Delete Vpc Endpoints

Description
 Delete Vpc Endpoints

Usage

ec2_delete_vpc_endpoints(
  VpcEndpointId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

VpcEndpointId List. One or more VPC endpoint IDs.
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
ec2_delete_vpc_endpoint_connection_notifications

Delete Vpc Endpoint Connection Notifications

Description

Deletes one or more VPC endpoint connection notifications.

Usage

c2_delete_vpc_endpoint_connection_notifications(
  ConnectionNotificationId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
ec2_delete_vpc_endpoint_service_configurations

Arguments

ConnectionNotificationId
List. One or more notification IDs.

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

simplify
Logical. Whether to simplify the result and handle nextToken in the response.[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value

A list object or a character vector

ConnectionNotificationId
One or more notification IDs.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description

Deletes one or more VPC endpoint service configurations in your account. Before you delete the endpoint service configuration, you must reject any Available or PendingAcceptance interface endpoint connections that are attached to the service.
Usage

```r
ec2_delete_vpc_endpoint_service_configurations(
    ServiceId, DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **ServiceId**: List. The IDs of one or more services.
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response[optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

- **ServiceId**: The IDs of one or more services.

- **DryRun**: Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`. 
Description

Deletes a VPC peering connection. Either the owner of the requester VPC or the owner of the accepter VPC can delete the VPC peering connection if it’s in the active state. The owner of the requester VPC can delete a VPC peering connection in the pending-acceptance state. You cannot delete a VPC peering connection that’s in the failed state.

Usage

```r
e2_delete_vpc_peering_connection(
  VpcPeeringConnectionId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **VpcPeeringConnectionId**
  - Character. The ID of the VPC peering connection.
- **DryRun**
  - Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
- **simplify**
  - Logical. Whether to simplify the result and handle `nextToken` in the response[optional]
- **others**
  - Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**
  - Logical. Whether to show an error message when a network error occurs.
- **retry_time**
  - Integer. Number of retries for a REST request when encountering the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**
  - Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.
- **region**
  - Character. The region of the AWS service.

Value

A list object or a character vector
VpcPeeringConnectionId
The ID of the VPC peering connection.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Describe
Delete Vpn Connection

Usage
```
ec2_delete_vpn_connection(
    VpnConnectionId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments
- **VpnConnectionId**
  Character. The ID of the VPN connection.
- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]
- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.
- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**
  Character. The region of the AWS service.
ec2_delete_vpn_connection_route

Value
A list object or a character vector

VpnConnectionId
The ID of the VPN connection.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Usage
```r
ec2_delete_vpn_connection_route(
    DestinationCidrBlock,
    VpnConnectionId,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments
- **DestinationCidrBlock**
  Character. The CIDR block associated with the local subnet of the customer network.

- **VpnConnectionId**
  Character. The ID of the VPN connection.

- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
ec2_delete_vpn_gateway

print_on_error  Logical. Whether to show an error message when a network error occurs.

retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  Character. The region of the AWS service.

Value

A list object or a character vector

DestinationCidrBlock

The CIDR block associated with the local subnet of the customer network.

VpnConnectionId

The ID of the VPN connection.

Description

Deletes the specified virtual private gateway. You must first detach the virtual private gateway from the VPC. Note that you don't need to delete the virtual private gateway if you plan to delete and recreate the VPN connection between your VPC and your network.

Usage

ec2_delete_vpn_gateway(
  VpnGatewayId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
ec2_deprovision_byoip_cidr

Arguments

- **VpnGatewayId**: Character. The ID of the virtual private gateway.
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request, [optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response [optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

**VpnGatewayId**

The ID of the virtual private gateway.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

---

**Description**

Deprovision Byoip Cidr
Usage

ec2_deprovision_byoip_cidr(
  Cidr,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

Cidr  Character. The address range, in CIDR notation.

DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]

simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]

others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.

Value

A list object or a character vector

Cidr

The address range, in CIDR notation. The prefix must be the same prefix that you specified when you provisioned the address range.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
Description
Deregister Image

Usage

```r
ec2_deregister_image(
  ImageId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **ImageId**
  Character. The ID of the AMI.

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  Character. The region of the AWS service.

Value

A list object or a character vector

**ImageId**

The ID of the AMI.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

**Description**

Deregisters tag keys to prevent tags that have the specified tag keys from being included in scheduled event notifications for resources in the Region.

**Usage**

```r
ec2_deregister_instance_event_notification_attributes(
  DryRun = NULL,
  InstanceTagAttribute = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

- **InstanceTagAttribute**
  Object. Information about the tag keys to deregister.[optional]

- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  Character. The region of the AWS service.
Value
A list object or a character vector

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

InstanceTagAttribute
Information about the tag keys to deregister.

deregister_transit_gateway_multicast_group_members
Deregister Transit Gateway Multicast Group Members

Description
Deregisters the specified members (network interfaces) from the transit gateway multicast group.

Usage
deregister_transit_gateway_multicast_group_members(
    TransitGatewayMulticastDomainId = NULL,
    GroupIpAddress = NULL,
    NetworkInterfaceIds = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments
TransitGatewayMulticastDomainId
    Character. The ID of the transit gateway multicast domain.[optional]
GroupIpAddress
    Character. The IP address assigned to the transit gateway multicast group.[optional]
NetworkInterfaceIds
    List. The IDs of the group members' network interfaces.[optional]
DryRun
    Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]
simplify
    Logical. Whether to simplify the result and handle nextToken in the response[optional]
**ec2_deregister_transit_gateway_multicast_group_sources**

Deregister Transit Gateway Multicast Group Sources

**Description**

Deregisters the specified sources (network interfaces) from the transit gateway multicast group.

**Value**

A list object or a character vector

**TransitGatewayMulticastDomainId**

The ID of the transit gateway multicast domain.

**GroupIpAddress**

The IP address assigned to the transit gateway multicast group.

**NetworkInterfaceIds**

The IDs of the group members' network interfaces.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
Usage

```r
ec2_deregister_transit_gateway_multicast_group_sources(
    TransitGatewayMulticastDomainId = NULL,
    GroupIpAddress = NULL,
    NetworkInterfaceIds = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **TransitGatewayMulticastDomainId**
  Character. The ID of the transit gateway multicast domain. [optional]

- **GroupIpAddress**
  Character. The IP address assigned to the transit gateway multicast group. [optional]

- **NetworkInterfaceIds**
  List. The IDs of the group sources’ network interfaces. [optional]

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]

- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response [optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  Character. The region of the AWS service.

Value

A list object or a character vector

**TransitGatewayMulticastDomainId**

The ID of the transit gateway multicast domain.

**GroupIpAddress**

The IP address assigned to the transit gateway multicast group.
NetworkInterfaceIds

The IDs of the group sources' network interfaces.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_describe_account_attributes

Describe Account Attributes

Description

Describe Account Attributes

Usage

ec2_describe_account_attributes(
    AttributeName = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

AttributeName List. The account attribute names.[optional]
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.
Value
A list object or a character vector

AttributeName
The account attribute names.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_describe_addresses
Describe Addresses

Description
Describe Addresses

Usage
ec2_describe_addresses(
    Filter = NULL,
    PublicIp = NULL,
    AllocationId = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments
Filter    Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]
PublicIp  List. One or more Elastic IP addresses. Default: Describes all your Elastic IP addresses. [optional]
AllocationId List. [EC2-VPC] Information about the allocation IDs.[optional]
DryRun    Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
simplify  Logical. Whether to simplify the result and handle nextToken in the response[optional]
ec2_describe_addresses

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value
A list object or a character vector

Filter
One or more filters. Filter names and values are case-sensitive.

• allocation-id - [EC2-VPC] The allocation ID for the address.
• association-id - [EC2-VPC] The association ID for the address.
• domain - Indicates whether the address is for use in EC2-Classic (standard) or in a VPC (vpc).
• instance-id - The ID of the instance the address is associated with, if any.
• network-border-group - A unique set of Availability Zones, Local Zones, or Wavelength Zones from where AWS advertises IP addresses.
• network-interface-id - [EC2-VPC] The ID of the network interface that the address is associated with, if any.
• network-interface-owner-id - The AWS account ID of the owner.
• private-ip-address - [EC2-VPC] The private IP address associated with the Elastic IP address.
• public-ip - The Elastic IP address, or the carrier IP address.
• tag:
: The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA for the filter value.
• tag-key - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.

PublicIp
One or more Elastic IP addresses.
Default: Describes all your Elastic IP addresses.
**AllocationId**

> [EC2-VPC] Information about the allocation IDs.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

---

**Description**

Describes the attributes of the specified Elastic IP addresses. For requirements, see Using reverse DNS for email applications.

**Usage**

```r
ec2_describe_addresses_attribute(
  AllocationId = NULL,
  Attribute = NULL,
  NextToken = NULL,
  MaxResults = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- **AllocationId**: List. [EC2-VPC] The allocation IDs.[optional]
- **Attribute**: Character. The attribute of the IP address.[optional]
- **NextToken**: Characters. The token for the next page of results[optional]
- **MaxResults**: Integer. The maximum number of results to return with a single call.[optional]
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.[optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response[optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error  Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region  Character. The region of the AWS service.

Value
A list object or a character vector

AllocationId
[EC2-VPC] The allocation IDs.

Attribute
The attribute of the IP address.

MaxResults
The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
Usage

```r
e2c_describe_aggregate_id_format(  
  DryRun = NULL,  
  simplify = TRUE,  
  others = list(),  
  print_on_error = aws_get_print_on_error(),  
  retry_time = aws_get_retry_time(),  
  network_timeout = aws_get_network_timeout(),  
  region = aws_get_region()
)
```

Arguments

- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.
Usage

```r
e2.describe_availability_zones(  
  Filter = NULL,  
  ZoneName = NULL,  
  ZoneId = NULL,  
  AllAvailabilityZones = NULL,  
  DryRun = NULL,  
  simplify = TRUE,  
  others = list(),  
  print_on_error = aws_get_print_on_error(),  
  retry_time = aws_get_retry_time(),  
  network_timeout = aws_get_network_timeout(),  
  region = aws_get_region()
)
```

Arguments

- **Filter**
  - Named list where the name is the filter name and the value is the filter's value.
  - The value can be a vector or a list object (see below)[optional]

- **ZoneName**
  - List. The names of the Availability Zones, Local Zones, and Wavelength Zones.[optional]

- **ZoneId**
  - List. The IDs of the Availability Zones, Local Zones, and Wavelength Zones.[optional]

- **AllAvailabilityZones**
  - Logical. Include all Availability Zones, Local Zones, and Wavelength Zones regardless of your opt-in status....[optional]

- **DryRun**
  - Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

- **simplify**
  - Logical. Whether to simplify the result and handle `nextToken` in the response[optional]

- **others**
  - Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  - Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  - Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  - Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  - Character. The region of the AWS service.

Value

A list object or a character vector

Filter

The filters.
- group-name - For Availability Zones, use the Region name. For Local Zones, use the name of the group associated with the Local Zone (for example, us-west-2-lax-1). For Wavelength Zones, use the name of the group associated with the Wavelength Zone (for example, us-east-1-wlz-bos-wlz-1).
- message - The Zone message.
- opt-in-status - The opt-in status (opted-in, and not-opted-in \ opt-in-not-required).
- parent-zoneID - The ID of the zone that handles some of the Local Zone and Wavelength Zone control plane operations, such as API calls.
- parent-zoneName - The ID of the zone that handles some of the Local Zone and Wavelength Zone control plane operations, such as API calls.
- region-name - The name of the Region for the Zone (for example, us-east-1).
- state - The state of the Availability Zone, the Local Zone, or the Wavelength Zone (available \ information \ impaired \ unavailable).
- zone-id - The ID of the Availability Zone (for example, use1-az1), the Local Zone (for example, usw2-1ax1-az1), or the Wavelength Zone (for example, us-east-1-wlz1-bos-wlz-1).
- zone-type - The type of zone, for example, local-zone.
- zone-name - The name of the Availability Zone (for example, us-east-1a), the Local Zone (for example, us-west-2-lax-1a), or the Wavelength Zone (for example, us-east-1-wlz1-bos-wlz-1).
- zone-type - The type of zone, for example, local-zone.

ZoneName

The names of the Availability Zones, Local Zones, and Wavelength Zones.

ZoneId

The IDs of the Availability Zones, Local Zones, and Wavelength Zones.

AllAvailabilityZones

Include all Availability Zones, Local Zones, and Wavelength Zones regardless of your opt-in status.

If you do not use this parameter, the results include only the zones for the Regions where you have chosen the option to opt in.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
ec2_describe_bundle_tasks

.describe Bundle Tasks

Description

Describe Bundle Tasks

Usage

ec2_describe_bundle_tasks(
  BundleId = NULL,
  Filter = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

BundleId        List. The bundle task IDs. Default: Describes all your bundle tasks. [optional]
Filter          Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]
DryRun          Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
simplify        Logical. Whether to simplify the result and handle nextToken in the response[optional]
others          Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error  Logical. Whether to show an error message when a network error occurs.
retry_time      Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region          Character. The region of the AWS service.

Value

A list object or a character vector
BundleId

The bundle task IDs.
Default: Describes all your bundle tasks.

Filter

The filters.

- bundle-id - The ID of the bundle task.
- error-code - If the task failed, the error code returned.
- error-message - If the task failed, the error message returned.
- instance-id - The ID of the instance.
- progress - The level of task completion, as a percentage (for example, 20%)
- s3-bucket - The Amazon S3 bucket to store the AMI.
- s3-prefix - The beginning of the AMI name.
- start-time - The time the task started (for example, 2013-09-15T17:15:20.000Z).
- state - The state of the task (pending | waiting-for-shutdown | bundling | storing | cancelling | complete | failed).
- update-time - The time of the most recent update for the task.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

description

Describe Byoip Cidrs

Usage

c2_describe_byoip_cidrs(MaxResults, DryRun = NULL, NextToken = NULL, simplify = TRUE, others = list(), print_on_error = aws_get_print_on_error(), retry_time = aws_get_retry_time(), network_timeout = aws_get_network_timeout(), region = aws_get_region())
Arguments

- **MaxResults**
  Integer. The maximum number of results to return with a single call.

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

- **NextToken**
  Characters. The token for the next page of results[optional]

- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  Character. The region of the AWS service.

Value

A list object or a character vector

**MaxResults**

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**ec2_describe_capacity_reservations**

*Describe Capacity Reservations*

**Description**

Describes one or more of your Capacity Reservations. The results describe only the Capacity Reservations in the AWS Region that you're currently using.
Usage

```r
ec2_describe_capacity_reservations(
    CapacityReservationId = NULL,
    NextToken = NULL,
    MaxResults = NULL,
    Filter = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **CapacityReservationId**:
  List. The ID of the Capacity Reservation. [optional]
- **NextToken**:
  Characters. The token for the next page of results. [optional]
- **MaxResults**:
  Integer. The maximum number of results to return for the request in a single page. [optional]
- **Filter**:
  Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below). [optional]
- **DryRun**:
  Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **simplify**:
  Logical. Whether to simplify the result and handle nextToken in the response. [optional]
- **others**:
  Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]
- **print_on_error**:
  Logical. Whether to show an error message when a network error occurs.
- **retry_time**:
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- **network_timeout**:
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**:
  Character. The region of the AWS service.

Value

A list object or a character vector

**CapacityReservationId**

The ID of the Capacity Reservation.
MaxResults

The maximum number of results to return for the request in a single page. The remaining results can be seen by sending another request with the returned nextToken value. This value can be between 5 and 500. If maxResults is given a larger value than 500, you receive an error.

Filter

One or more filters.

- **instance-type** - The type of instance for which the Capacity Reservation reserves capacity.
- **owner-id** - The ID of the AWS account that owns the Capacity Reservation.
- **availability-zone-id** - The Availability Zone ID of the Capacity Reservation.
- **instance-platform** - The type of operating system for which the Capacity Reservation reserves capacity.
- **availability-zone** - The Availability Zone ID of the Capacity Reservation.
- **tenancy** - Indicates the tenancy of the Capacity Reservation. A Capacity Reservation can have one of the following tenancy settings:
  - **default** - The Capacity Reservation is created on hardware that is shared with other AWS accounts.
  - **dedicated** - The Capacity Reservation is created on single-tenant hardware that is dedicated to a single AWS account.
- **state** - The current state of the Capacity Reservation. A Capacity Reservation can be in one of the following states:
  - **active** - The Capacity Reservation is active and the capacity is available for your use.
  - **expired** - The Capacity Reservation expired automatically at the date and time specified in your request. The reserved capacity is no longer available for your use.
  - **cancelled** - The Capacity Reservation was cancelled. The reserved capacity is no longer available for your use.
  - **pending** - The Capacity Reservation request was successful but the capacity provisioning is still pending.
  - **failed** - The Capacity Reservation request has failed. A request might fail due to invalid request parameters, capacity constraints, or instance limit constraints. Failed requests are retained for 60 minutes.
- **end-date** - The date and time at which the Capacity Reservation expires. When a Capacity Reservation expires, the reserved capacity is released and you can no longer launch instances into it. The Capacity Reservation’s state changes to expired when it reaches its end date and time.
- **end-date-type** - Indicates the way in which the Capacity Reservation ends. A Capacity Reservation can have one of the following end types:
  - **unlimited** - The Capacity Reservation remains active until you explicitly cancel it.
  - **limited** - The Capacity Reservation expires automatically at a specified date and time.
- **instance-match-criteria** - Indicates the type of instance launches that the Capacity Reservation accepts. The options include:
– open - The Capacity Reservation accepts all instances that have matching attributes (instance type, platform, and Availability Zone). Instances that have matching attributes launch into the Capacity Reservation automatically without specifying any additional parameters.

– targeted - The Capacity Reservation only accepts instances that have matching attributes (instance type, platform, and Availability Zone), and explicitly target the Capacity Reservation. This ensures that only permitted instances can use the reserved capacity.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is *DryRunOperation*. Otherwise, it is *UnauthorizedOperation*.

---

**ec2_describe_carrier_gateways**

*Describe Carrier Gateways*

**Description**

Describes one or more of your carrier gateways.

**Usage**

```r
ec2_describe_carrier_gateways(
    CarrierGatewayId = NULL,
    Filter = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

**Arguments**

- **CarrierGatewayId**
  - List. One or more carrier gateway IDs.[optional]
- **Filter**
  - Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below).[optional]
- **MaxResults**
  - Integer. The maximum number of results to return with a single call.[optional]
- **NextToken**
  - Characters. The token for the next page of results[optional]
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value
A list object or a character vector

CarrierGatewayId
One or more carrier gateway IDs.

Filter
One or more filters.

- carrier-gateway-id - The ID of the carrier gateway.
- state - The state of the carrier gateway (pending \| failed \| available \| deleting \| deleted).
- owner-id - The AWS account ID of the owner of the carrier gateway.
- tag:\ - The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA for the filter value.
- tag-key - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.
- vpc-id - The ID of the VPC associated with the carrier gateway.

MaxResults
The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
ec2_describe_classic_link_instances

Describe Classic Link Instances

Description

Describes one or more of your linked EC2-Classic instances. This request only returns information about EC2-Classic instances linked to a VPC through ClassicLink. You cannot use this request to return information about other instances.

Usage

ec2_describe_classic_link_instances(
    Filter = NULL,
    DryRun = NULL,
    InstanceId = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

Filter  Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]
DryRun  Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
InstanceId  List. One or more instance IDs. Must be instances linked to a VPC through ClassicLink.[optional]
MaxResults  Integer. The maximum number of results to return with a single call.[optional]
NextToken  Characters. The token for the next page of results[optional]
simplify  Logical. Whether to simplify the result and handle nextToken in the response[optional]
others  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error  Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout

   Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region

   Character. The region of the AWS service.

Value

   A list object or a character vector

Filter

   One or more filters.

   - group-id - The ID of a VPC security group that\'s associated with the instance.
   - instance-id - The ID of the instance.
   - tag\: The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA for the filter value.
   - tag-key - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.
   - vpc-id - The ID of the VPC to which the instance is linked.

DryRun

   Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

InstanceId

   One or more instance IDs. Must be instances linked to a VPC through ClassicLink.

MaxResults

   The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

   Constraint: If the value is greater than 1000, we return only 1000 items.
Describe Client Vpn Authorization Rules

Description

Describes the authorization rules for a specified Client VPN endpoint.

Usage

ec2_describe_client_vpn_authorization_rules(
    ClientVpnEndpointId,
    DryRun = NULL,
    NextToken = NULL,
    Filter = NULL,
    MaxResults = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

- **ClientVpnEndpointId**: Character. The ID of the Client VPN endpoint.
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
- **NextToken**: Characters. The token for the next page of results.[optional]
- **Filter**: Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below).[optional]
- **MaxResults**: Integer. The maximum number of results to return for the request in a single page.[optional]
- **simplify**: Logical. Whether to simplify the result and handle NextToken in the response.[optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request.[optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.
Value
A list object or a character vector

ClientVpnEndpointId
The ID of the Client VPN endpoint.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Filter
One or more filters. Filter names and values are case-sensitive.
- description - The description of the authorization rule.
- destination-cidr - The CIDR of the network to which the authorization rule applies.
- group-id - The ID of the Active Directory group to which the authorization rule grants access.

MaxResults
The maximum number of results to return for the request in a single page. The remaining results can be seen by sending another request with the nextToken value.

---

```r
ec2_describe_client_vpn_connections

Describe Client Vpn Connections

Description
Describes active client connections and connections that have been terminated within the last 60 minutes for the specified Client VPN endpoint.

Usage
ec2_describe_client_vpn_connections(
  ClientVpnEndpointId,
  Filter = NULL,
  NextToken = NULL,
  MaxResults = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time()
)```
network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

ClientVpnEndpointId
    Character. The ID of the Client VPN endpoint.

Filter
    Named list where the name is the filter name and the value is the filter's value.
The value can be a vector or a list object (see below)[optional]

NextToken
    Characters. The token for the next page of results[optional]

MaxResults
    Integer. The maximum number of results to return for the request in a single page.[optional]

DryRun
    Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

simplify
    Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
    Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
    Logical. Whether to show an error message when a network error occurs.

retry_time
    Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
    Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
    Character. The region of the AWS service.

Value

A list object or a character vector

ClientVpnEndpointId

The ID of the Client VPN endpoint.

Filter

One or more filters. Filter names and values are case-sensitive.

- connection-id - The ID of the connection.
- username - For Active Directory client authentication, the user name of the client who established the client connection.

MaxResults

The maximum number of results to return for the request in a single page. The remaining results can be seen by sending another request with the nextToken value.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

```
ec2_describe_client_vpn_endpoints

Describe Client Vpn Endpoints

Description

Describes one or more Client VPN endpoints in the account.

Usage

ec2_describe_client_vpn_endpoints(
    ClientVpnEndpointId = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    Filter = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

ClientVpnEndpointId
    List. The ID of the Client VPN endpoint.[optional]

MaxResults
    Integer. The maximum number of results to return for the request in a single page.[optional]

NextToken
    Characters. The token for the next page of results[optional]

Filter
    Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]

DryRun
    Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

simplify
    Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
    Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
    Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  Character. The region of the AWS service.

**Value**

A list object or a character vector

**ClientVpnEndpointId**

The ID of the Client VPN endpoint.

**MaxResults**

The maximum number of results to return for the request in a single page. The remaining results can be seen by sending another request with the nextToken value.

**Filter**

One or more filters. Filter names and values are case-sensitive.

- endpoint-id - The ID of the Client VPN endpoint.
- transport-protocol - The transport protocol (tcp \| udp).

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
Usage

```r
ec2_describe_client_vpn_routes(
    ClientVpnEndpointId,
    Filter = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **ClientVpnEndpointId**
  Character. The ID of the Client VPN endpoint.

- **Filter**
  Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]

- **MaxResults**
  Integer. The maximum number of results to return for the request in a single page.[optional]

- **NextToken**
  Characters. The token for the next page of results[optional]

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  Character. The region of the AWS service.

Value

A list object or a character vector

**ClientVpnEndpointId**

The ID of the Client VPN endpoint.
Filter

One or more filters. Filter names and values are case-sensitive.

- destination-cidr - The CIDR of the route destination.
- origin - How the route was associated with the Client VPN endpoint (associate \ add-route).
- target-subnet - The ID of the subnet through which traffic is routed.

MaxResults

The maximum number of results to return for the request in a single page. The remaining results can be seen by sending another request with the nextToken value.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_describe_client_vpn_target_networks

Describe Client Vpn Target Networks

Description

Describes the target networks associated with the specified Client VPN endpoint.

Usage

```r
ec2_describe_client_vpn_target_networks(
    ClientVpnEndpointId,
    AssociationIds = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    Filter = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```
Arguments

ClientVpnEndpointId
Character. The ID of the Client VPN endpoint.

AssociationIds
List. The IDs of the target network associations.[optional]

MaxResults
Integer. The maximum number of results to return for the request in a single page.[optional]

NextToken
Characters. The token for the next page of results[optional]

Filter
Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value
A list object or a character vector

ClientVpnEndpointId
The ID of the Client VPN endpoint.

AssociationIds
The IDs of the target network associations.

MaxResults
The maximum number of results to return for the request in a single page. The remaining results can be seen by sending another request with the nextToken value.

Filter
One or more filters. Filter names and values are case-sensitive.

- association-id - The ID of the association.
- target-network-id - The ID of the subnet specified as the target network.
- vpc-id - The ID of the VPC in which the target network is located.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_describe_coip_pools

Describe Coip Pools

Description

Describes the specified customer-owned address pools or all of your customer-owned address pools.

Usage

ec2_describe_coip_pools(
    PoolId = NULL,
    Filter = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

PoolId List. The IDs of the address pools.[optional]
Filter Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]
MaxResults Integer. The maximum number of results to return with a single call.[optional]
NextToken Characters. The token for the next page of results[optional]
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
ec2_describe_conversion_tasks

network_timeout
   Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
   Character. The region of the AWS service.

Value
   A list object or a character vector

PoolId
   The IDs of the address pools.

Filter
   The filters. The following are the possible values:

      • coip-pool.pool-id

      • coip-pool.local-gateway-route-table-id

MaxResults
   The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

DryRun
   Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_describe_conversion_tasks

Describe Conversion Tasks

Description
   Describe Conversion Tasks
ec2_describe_conversion_tasks

Usage

```r
ec2_describe_conversion_tasks(
    ConversionTaskId = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **ConversionTaskId**
  List. The conversion task IDs.

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request.

- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response.

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request.

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  Character. The region of the AWS service.

Value

A list object or a character vector

**ConversionTaskId**

The conversion task IDs.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.  


ec2_describe_customer_gateways

Description

Describe Customer Gateways

Usage

describe_customer_gateways(
  CustomerGatewayId = NULL,
  Filter = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

CustomerGatewayId
  List. One or more customer gateway IDs. Default: Describes all your customer gateways. [optional]

Filter
  Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]

DryRun
  Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]

simplify
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
  Logical. Whether to show an error message when a network error occurs.

retry_time
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
  Character. The region of the AWS service.

Value

A list object or a character vector
CustomerGatewayId

One or more customer gateway IDs.
Default: Describes all your customer gateways.

Filter

One or more filters.

- bgp-asn - The customer gateway's Border Gateway Protocol (BGP) Autonomous System Number (ASN).
- customer-gateway-id - The ID of the customer gateway.
- ip-address - The IP address of the customer gateway's Internet-routable external interface.
- state - The state of the customer gateway (pending | available | deleting | deleted).
- type - The type of customer gateway. Currently, the only supported type is ipsec.1.
- tag: - The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA for the filter value.
- tag-key - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

DhcpOptionsId List. The IDs of one or more DHCP options sets. Default: Describes all your DHCP options sets. [optional]
Filter Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
NextToken Characters. The token for the next page of results[optional]
MaxResults Integer. The maximum number of results to return with a single call.[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.

Value

A list object or a character vector

DhcpOptionsId

The IDs of one or more DHCP options sets.
Default: Describes all your DHCP options sets.

Filter

One or more filters.

• dhcp-options-id - The ID of a DHCP options set.
• key - The key for one of the options (for example, domain-name).
• value - The value for one of the options.
• owner-id - The ID of the AWS account that owns the DHCP options set.
• tag: - The key/value combination of a tag assigned to the resource. Use the tag key in the
filter name and the tag value as the filter value. For example, to find all resources that have a
tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA
for the filter value.
• tag-key - The key of a tag assigned to the resource. Use this filter to find all resources
assigned a tag with a specific key, regardless of the tag value.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the
request, and provides an error response. If you have the required permissions, the error response is
DryRunOperation. Otherwise, it is UnauthorizedOperation.

**MaxResults**

The maximum number of results to return with a single call. To retrieve the remaining results, make
another call with the returned nextToken value.

---

**ec2_describe_egress_only_internet_gateways**

*Describe Egress Only Internet Gateways*

**Description**

Describes one or more of your egress-only internet gateways.

**Usage**

```r
ec2_describe_egress_only_internet_gateways(
  DryRun = NULL,
  EgressOnlyInternetGatewayId = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  Filter = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- **DryRun** Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
EgressOnlyInternetGatewayId
List. One or more egress-only internet gateway IDs.[optional]

MaxResults
Integer. The maximum number of results to return with a single call.[optional]

NextToken
Characters. The token for the next page of results[optional]

Filter
Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value
A list object or a character vector

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

EgressOnlyInternetGatewayId
One or more egress-only internet gateway IDs.

MaxResults
The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

Filter
One or more filters.

• tag:val - The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA for the filter value.

• tag-key - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.
ec2_describe_elastic_gpus

Describe Elastic Gpus

Description
Describes the Elastic Graphics accelerator associated with your instances. For more information about Elastic Graphics, see Amazon Elastic Graphics.

Usage
ec2_describe_elastic_gpus(
    ElasticGpuId = NULL,
    DryRun = NULL,
    Filter = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ElasticGpuId</td>
<td>List. The Elastic Graphics accelerator IDs.[optional]</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]</td>
</tr>
<tr>
<td>Filter</td>
<td>Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]</td>
</tr>
<tr>
<td>MaxResults</td>
<td>Integer. The maximum number of results to return in a single call.[optional]</td>
</tr>
<tr>
<td>NextToken</td>
<td>Characters. The token for the next page of results[optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response[optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>
Value

A list object or a character vector

ElasticGpuId

The Elastic Graphics accelerator IDs.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Filter

The filters.

- availability-zone - The Availability Zone in which the Elastic Graphics accelerator resides.
- elastic-gpu-type - The type of Elastic Graphics accelerator; for example, eg1.medium.
- instance-id - The ID of the instance to which the Elastic Graphics accelerator is associated.

MaxResults

The maximum number of results to return in a single call. To retrieve the remaining results, make another call with the returned NextToken value. This value can be between 5 and 1000.

---

**ec2_describe_export_image_tasks**

*Describe Export Image Tasks*

---

**Description**

Describes the specified export image tasks or all of your export image tasks.

**Usage**

```r
ec2_describe_export_image_tasks(
  DryRun = NULL,
  Filter = NULL,
  ExportImageTaskId = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  simplify = TRUE,
)```
others = list(),
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request, [optional]
Filter Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below) [optional]
ExportImageTaskId List. The IDs of the export image tasks. [optional]
MaxResults Integer. The maximum number of results to return in a single call. [optional]
NextToken Characters. The token for the next page of results. [optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response. [optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value
A list object or a character vector

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Filter
Filter tasks using the task-state filter and one of the following values: active, completed, deleting, or deleted.

ExportImageTaskId
The IDs of the export image tasks.
MaxResults

The maximum number of results to return in a single call.

---

**ec2_describe_export_tasks**

*Describe Export Tasks*

---

**Description**

Describes the specified export instance tasks or all of your export instance tasks.

**Usage**

```
ec2_describe_export_tasks(
    ExportTaskId = NULL,
    Filter = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

**Arguments**

- **ExportTaskId** List. The export task IDs.[optional]
- **Filter** Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]
- **simplify** Logical. Whether to simplify the result and handle `nextToken` in the response[optional]
- **others** Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error** Logical. Whether to show an error message when a network error occurs.
- **retry_time** Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout** Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region** Character. The region of the AWS service.

**Value**

A list object or a character vector
ec2_describe_fast_snapshot_restores

ExportTaskId

The export task IDs.

Filter

the filters for the export tasks.

describe_fast_snapshot_restores

Describe Fast Snapshot Restores

Description

Describes the state of fast snapshot restores for your snapshots.

Usage

cy2_describe_fast_snapshot_restores(
    Filter = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

Filter Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]
MaxResults Integer. The maximum number of results to return with a single call.[optional]
NextToken Characters. The token for the next page of results[optional]
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout
   Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
   Character. The region of the AWS service.

Value
   A list object or a character vector

Filter
   The filters. The possible values are:
   • availability-zone: The Availability Zone of the snapshot.
   • owner-id: The ID of the AWS account that enabled fast snapshot restore on the snapshot.
   • snapshot-id: The ID of the snapshot.
   • state: The state of fast snapshot restores for the snapshot (enabling \| optimizing \| enabled \| disabling \| disabled).

MaxResults
   The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

DryRun
   Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

ec2_describe_fleets  Describe Fleets

Description
   Describe Fleets

Usage
   ec2_describe_fleets(
      DryRun = NULL,
      MaxResults = NULL,
      NextToken = NULL,
      FleetId = NULL,
      Filter = NULL,
      simplify = TRUE,
      others = list(),
   )
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]
MaxResults Integer. The maximum number of results to return in a single call.[optional]
NextToken Characters. The token for the next page of results[optional]
FleetId List. The ID of the EC2 Fleets.[optional]
Filter Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value

A list object or a character vector

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

MaxResults

The maximum number of results to return in a single call. Specify a value between 1 and 1000. The default value is 1000. To retrieve the remaining results, make another call with the returned NextToken value.

FleetId

The ID of the EC2 Fleets.
Filter
The filters.

- activity-status - The progress of the EC2 Fleet (error \| pending-fulfillment \| pending-termination \| fulfilled).
- excess-capacity-termination-policy - Indicates whether to terminate running instances if the target capacity is decreased below the current EC2 Fleet size (true \| false).
- fleet-state - The state of the EC2 Fleet (submitted \| active \| deleted \| failed \| deleted-running \| deleted-terminating \| modifying).
- replace-unhealthy-instances - Indicates whether EC2 Fleet should replace unhealthy instances (true \| false).
- type - The type of request (instant \| request \| maintain).

---

e2_describe_fleet_history

Describe Fleet History

Description
Describe Fleet History

Usage
e2_describe_fleet_history(
    FleetId,
    StartTime,
    DryRun = NULL,
    EventType = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

FleetId Character. The ID of the EC2 Fleet.
StartTime Character. The start date and time for the events, in UTC format (for example, YYYY-MM-DDTHH:MM:SSZ)....
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]
**eventType**  Character. The type of events to describe. By default, all events are described. [optional]

**maxResults**  Integer. The maximum number of results to return in a single call. [optional]

**nextToken**  Characters. The token for the next page of results. [optional]

**simplify**  Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]

**others**  Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]

**print_on_error**  Logical. Whether to show an error message when a network error occurs.

**retry_time**  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

**network_timeout**  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

**region**  Character. The region of the AWS service.

**Value**

A list object or a character vector

**FleetId**

The ID of the EC2 Fleet.

**StartTime**

The start date and time for the events, in UTC format (for example, `YYYY-MM-DDTHH:MM:SSZ`).

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

**eventType**

The type of events to describe. By default, all events are described.

**MaxResults**

The maximum number of results to return in a single call. Specify a value between 1 and 1000. The default value is 1000. To retrieve the remaining results, make another call with the returned `NextToken` value.
ec2_describe_fleet_instances

Description

Describe Fleet Instances

Usage

ec2_describe_fleet_instances(
  FleetId,
  DryRun = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  Filter = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FleetId</td>
<td>Character. The ID of the EC2 Fleet.</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action,</td>
</tr>
<tr>
<td></td>
<td>without actually making the request.</td>
</tr>
<tr>
<td>MaxResults</td>
<td>Integer. The maximum number of results to return in a single call.</td>
</tr>
<tr>
<td>NextToken</td>
<td>Characters. The token for the next page of results</td>
</tr>
<tr>
<td>Filter</td>
<td>Named list where the name is the filter name and the value is the filter's</td>
</tr>
<tr>
<td></td>
<td>value. The value can be a vector or a list object (see below).</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters</td>
</tr>
<tr>
<td></td>
<td>and need to be added into the request.</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network</td>
</tr>
<tr>
<td></td>
<td>issue. If the request has been sent retry_time times but still not be able</td>
</tr>
<tr>
<td></td>
<td>to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can</td>
</tr>
<tr>
<td></td>
<td>not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>
Value
   A list object or a character vector

FleetId
   The ID of the EC2 Fleet.

DryRun
   Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

MaxResults
   The maximum number of results to return in a single call. Specify a value between 1 and 1000. The default value is 1000. To retrieve the remaining results, make another call with the returned NextToken value.

Filter
   The filters.
   * instance-type - The instance type.

---

ec2_describe_flow_logs

Describe Flow Logs

Description

Describes one or more flow logs. To view the information in your flow logs (the log streams for the network interfaces), you must use the CloudWatch Logs console or the CloudWatch Logs API.

Usage

```r
ec2_describe_flow_logs(
   DryRun = NULL,
   Filter = NULL,
   FlowLogId = NULL,
   MaxResults = NULL,
   NextToken = NULL,
   simplify = TRUE,
   others = list(),
   print_on_error = aws_get_print_on_error(),
   retry_time = aws_get_retry_time(),
   network_timeout = aws_get_network_timeout(),
   region = aws_get_region()
)
```
Arguments

DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request, [optional]

Filter Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below) [optional]

FlowLogId List. One or more flow log IDs. Constraint: Maximum of 1000 flow log IDs. [optional]

MaxResults Integer. The maximum number of results to return with a single call. [optional]

NextToken Characters. The token for the next page of results [optional]

simplify Logical. Whether to simplify the result and handle nextToken in the response [optional]

others Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.

Value

A list object or a character vector

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Filter

One or more filters.

- **deliver-log-status** - The status of the logs delivery (SUCCESS \ SUCCESS \ FAILED).
- **log-destination-type** - The type of destination to which the flow log publishes data. Possible destination types include cloud-watch-logs and s3.
- **flow-log-id** - The ID of the flow log.
- **log-group-name** - The name of the log group.
- **resource-id** - The ID of the VPC, subnet, or network interface.
- **traffic-type** - The type of traffic (ACCEPT \ REJECT \ ALL).
• **tag**: The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key **Owner** and the value **TeamA**, specify **tag:Owner** for the filter name and **TeamA** for the filter value.

• **tag-key**: The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.

**FlowLogId**

One or more flow log IDs.

Constraint: Maximum of 1000 flow log IDs.

**MaxResults**

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned **nextToken** value.

```r
ec2_describe_fpga_images(  
  DryRun = NULL,  
  FpgaImageId = NULL,  
  Owner = NULL,  
  Filter = NULL,  
  NextToken = NULL,  
  MaxResults = NULL,  
  simplify = TRUE,  
  others = list(),  
  print_on_error = aws_get_print_on_error(),  
  retry_time = aws_get_retry_time(),  
  network_timeout = aws_get_network_timeout(),  
  region = aws_get_region())
```
Arguments

DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]

FpgaImageId List. The AFI IDs. [optional]

Owner List. Filters the AFI by owner. [optional]

Filter Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below). [optional]

NextToken Characters. The token for the next page of results. [optional]

MaxResults Integer. The maximum number of results to return in a single call. [optional]

simplify Logical. Whether to simplify the result and handle nextToken in the response. [optional]

others Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.

Value

A list object or a character vector

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

FpgaImageId

The AFI IDs.

Owner

Filters the AFI by owner. Specify an AWS account ID, self (owner is the sender of the request), or an AWS owner alias (valid values are amazon \ aws-marketplace).

Filter

The filters.

• create-time - The creation time of the AFI.
• fpga-image-id - The FPGA image identifier (AFI ID).
• fpga-image-global-id - The global FPGA image identifier (AGFI ID).
• name - The name of the AFI.
• owner-id - The AWS account ID of the AFI owner.
• product-code - The product code.
• shell-version - The version of the AWS Shell that was used to create the bitstream.
• state - The state of the AFI (pending | failed | available | unavailable).
• tag:\ - The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA for the filter value.
• tag-key - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.
• update-time - The time of the most recent update.

MaxResults

The maximum number of results to return in a single call.

---

**ec2_describe_fpga_image_attribute**

*Describe Fpga Image Attribute*

---

**Description**

Describes the specified attribute of the specified Amazon FPGA Image (AFI).

**Usage**

```r
ec2_describe_fpga_image_attribute(
  FpgaImageId,
  Attribute,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
Arguments

FpgaImageId   Character. The ID of the AFI.
Attribute     Character. The AFI attribute.
DryRun        Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
simplify     Logical. Whether to simplify the result and handle nextToken in the response[optional]
others       Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time   Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region       Character. The region of the AWS service.

Value

A list object or a character vector

FpgaImageId

The ID of the AFI.

Attribute

The AFI attribute.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_describe_hosts    Describe Hosts

Description

Describe Hosts
Usage

```r
call <- ec2_describe_hosts(
  Filter = NULL,
  HostId = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **Filter**: Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below) [optional]
- **HostId**: List. The IDs of the Dedicated Hosts. The IDs are used for targeted instance launches [optional]
- **MaxResults**: Integer. The maximum number of results to return for the request in a single page [optional]
- **NextToken**: Characters. The token for the next page of results [optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response [optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

Filter

The filters.

- `auto-placement` - Whether auto-placement is enabled or disabled (on \ off).
- `availability-zone` - The Availability Zone of the host.
- `client-token` - The idempotency token that you provided when you allocated the host.
• **host-reservation-id** - The ID of the reservation assigned to this host.

• **instance-type** - The instance type size that the Dedicated Host is configured to support.

• **state** - The allocation state of the Dedicated Host (available | under-assessment | permanent-failure | released | released-permanent-failure).

• **tag-key** - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.

**HostId**

The IDs of the Dedicated Hosts. The IDs are used for targeted instance launches.

**MaxResults**

The maximum number of results to return for the request in a single page. The remaining results can be seen by sending another request with the returned `nextToken` value. This value can be between 5 and 500. If `maxResults` is given a larger value than 500, you receive an error.

You cannot specify this parameter and the host IDs parameter in the same request.

**Usage**

```r
ec2_describe_host_reservations(
  Filter = NULL,
  HostReservationIdSet = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
Arguments

Filter
Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]

HostReservationIdSet
List. The host reservation IDs.[optional]

MaxResults
Integer. The maximum number of results to return for the request in a single page.[optional]

NextToken
Characters. The token for the next page of results[optional]

simplify
Logical. Whether to simplify the result and handle NextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encountering the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.

region
Character. The region of the AWS service.

Value

A list object or a character vector

Filter

The filters.

- instance-family - The instance family (for example, m4).
- payment-option - The payment option (NoUpfront \ PartialUpfront \ AllUpfront).
- state - The state of the reservation (payment-pending \ payment-failed \ active \ retired).
- tag: - The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA for the filter value.
- tag-key - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.

HostReservationIdSet

The host reservation IDs.

MaxResults

The maximum number of results to return for the request in a single page. The remaining results can be seen by sending another request with the returned NextToken value. This value can be between 5 and 500. If maxResults is given a larger value than 500, you receive an error.
ec2\_describe\_host\_reservation\_offerings

*Describe Host Reservation Offerings*

**Description**

Describe Host Reservation Offerings

**Usage**

```r
ec2\_describe\_host\_reservation\_offerings(
  Filter = NULL,
  MaxDuration = NULL,
  MaxResults = NULL,
  MinDuration = NULL,
  NextToken = NULL,
  OfferingId = NULL,
  simplify = TRUE,
  others = list(),
  print\_on\_error = aws\_get\_print\_on\_error(),
  retry\_time = aws\_get\_retry\_time(),
  network\_timeout = aws\_get\_network\_timeout(),
  region = aws\_get\_region()
)
```

**Arguments**

- **Filter**  
  Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]
- **MaxDuration**  
  Integer. This is the maximum duration of the reservation to purchase, specified in seconds.[optional]
- **MaxResults**  
  Integer. The maximum number of results to return for the request in a single page.[optional]
- **MinDuration**  
  Integer. This is the minimum duration of the reservation you'd like to purchase, specified in seconds.[optional]
- **NextToken**  
  Characters. The token for the next page of results[optional]
- **OfferingId**  
  Character. The ID of the reservation offering.[optional]
- **simplify**  
  Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others**  
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print\_on\_error**  
  Logical. Whether to show an error message when a network error occurs.
- **retry\_time**  
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry\_time times but still not be able to get the response, an error will be thrown.
network_timeout
   Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
   Character. The region of the AWS service.

Value
   A list object or a character vector

Filter
   The filters.
   - instance-family - The instance family of the offering (for example, m4).
   - payment-option - The payment option (NoUpfront | PartialUpfront | AllUpfront).

MaxDuration
   This is the maximum duration of the reservation to purchase, specified in seconds. Reservations are available in one-year and three-year terms. The number of seconds specified must be the number of seconds in a year (365x24x60x60) times one of the supported durations (1 or 3). For example, specify 94608000 for three years.

MaxResults
   The maximum number of results to return for the request in a single page. The remaining results can be seen by sending another request with the returned nextToken value. This value can be between 5 and 500. If maxResults is given a larger value than 500, you receive an error.

MinDuration
   This is the minimum duration of the reservation you'd like to purchase, specified in seconds. Reservations are available in one-year and three-year terms. The number of seconds specified must be the number of seconds in a year (365x24x60x60) times one of the supported durations (1 or 3). For example, specify 31536000 for one year.

OfferingId
   The ID of the reservation offering.
ec2_describe_iam_instance_profile_associations

Describe Iam Instance Profile Associations

Description

Describes your IAM instance profile associations.

Usage

ec2_describe_iam_instance_profile_associations(
  AssociationId = NULL,
  Filter = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

AssociationId List. The IAM instance profile associations,[optional]
Filter Named list where the name is the filter name and the value is the filter’s value.
The value can be a vector or a list object (see below)[optional]
MaxResults Integer. The maximum number of results to return in a single call.[optional]
NextToken Characters. The token for the next page of results[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and
need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network
issue. If the request has been sent retry_time times but still not be able to get
the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can
not be less than 1 ms.
region Character. The region of the AWS service.

Value

A list object or a character vector
AssociationId

The IAM instance profile associations.

Filter

The filters.

• instance-id - The ID of the instance.
• state - The state of the association (associating \| associated \| disassociating).

MaxResults

The maximum number of results to return in a single call. To retrieve the remaining results, make another call with the returned NextToken value.

ec2_describe_identity_id_format

Describe Identity Id Format

Usage

ec2_describe_identity_id_format(
  PrincipalArn,
  Resource = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

PrincipalArn Character. The ARN of the principal, which can be an IAM role, IAM user, or the root user.
Resource Character. The type of resource: bundle \| conversion-task \| customer-gateway \| dhcp-options \| elastic-ip-allocation...[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
Describe Id Format

**Description**
Describe Id Format

**Usage**
```r
e2_describe_id_format(
    Resource = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```
**Arguments**

<table>
<thead>
<tr>
<th>Resource</th>
<th>Character. The type of resource: bundle</th>
<th>conversion-task</th>
<th>customer-gateway</th>
<th>dhcp-options</th>
<th>elastic-ip-allocation...[optional]</th>
</tr>
</thead>
<tbody>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response[optional]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Value**

A list object or a character vector

**Resource**


---

**ec2_describe_images**  
*Describe Images*

**Description**

Describe Images

**Usage**

```r
e2c_describe_images(  
  ExecutableBy = NULL,  
  Filter = NULL,  
  ImageId = NULL,  
  Owner = NULL,  
  DryRun = NULL,  
  simplify = TRUE,  
  others = list(),  
)```
Arguments

ExecutableBy List. Scopes the images by users with explicit launch permissions. [optional]
Filter Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]
ImageId List. The image IDs. Default: Describes all images available to you. [optional]
Owner List. Scopes the results to images with the specified owners.[optional]
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value

A list object or a character vector

ExecutableBy

Scopes the images by users with explicit launch permissions. Specify an AWS account ID, self (the sender of the request), or all (public AMIs).

Filter

The filters.

- architecture - The image architecture (i386 | x86_64 | arm64).
- block-device-mapping.delete-on-termination - A Boolean value that indicates whether the Amazon EBS volume is deleted on instance termination.
- block-device-mapping.device-name - The device name specified in the block device mapping (for example, /dev/sdh or xvdh).
- block-device-mapping.snapshot-id - The ID of the snapshot used for the EBS volume.
• block-device-mapping.volume-size - The volume size of the EBS volume, in GiB.
• block-device-mapping.volume-type - The volume type of the EBS volume (gp2 | io1 | io2 | st1 | sc1 | standard).
• block-device-mapping.encrypted - A Boolean that indicates whether the EBS volume is encrypted.
• description - The description of the image (provided during image creation).
• ena-support - A Boolean that indicates whether enhanced networking with ENA is enabled.
• hypervisor - The hypervisor type (ovm | xen).
• image-id - The ID of the image.
• image-type - The image type (machine | kernel | ramdisk).
• is-public - A Boolean that indicates whether the image is public.
• kernel-id - The kernel ID.
• manifest-location - The location of the image manifest.
• name - The name of the AMI (provided during image creation).
• owner-alias - The owner alias (amazon | aws-marketplace). The valid aliases are defined in an Amazon-maintained list. This is not the AWS account alias that can be set using the IAM console. We recommend that you use the Owner request parameter instead of this filter.
• owner-id - The AWS account ID of the owner. We recommend that you use the Owner request parameter instead of this filter.
• platform - The platform. To only list Windows-based AMIs, use windows.
• product-code - The product code.
• product-code.type - The type of the product code (devpay | marketplace).
• ramdisk-id - The RAM disk ID.
• root-device-name - The device name of the root device volume (for example, /dev/sda1).
• root-device-type - The type of the root device volume (ebs | instance-store).
• state - The state of the image (available | pending | failed).
• state-reason-code - The reason code for the state change.
• state-reason-message - The message for the state change.
• sriov-net-support - A value of simple indicates that enhanced networking with the Intel 82599 VF interface is enabled.
• tag: - The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA for the filter value.
• tag-key - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.
• virtualization-type - The virtualization type (paravirtual | hvm).

ImageId
The image IDs.
Default: Describes all images available to you.
Owner
Scopes the results to images with the specified owners. You can specify a combination of AWS account IDs, self, amazon, and aws-marketplace. If you omit this parameter, the results include all images for which you have launch permissions, regardless of ownership.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_describe_image_attribute

Description
Describes the specified attribute of the specified AMI. You can specify only one attribute at a time.

Usage
ect2_describe_image_attribute(
    Attribute,
    ImageId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attribute</td>
<td>Character. The AMI attribute.</td>
</tr>
<tr>
<td>ImageId</td>
<td>Character. The ID of the AMI.</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response.[optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request.[optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
</tr>
</tbody>
</table>
network_timeout
    Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
    Character. The region of the AWS service.

Value
    A list object or a character vector

Attribute
    The AMI attribute.
    Note: The blockDeviceMapping attribute is deprecated. Using this attribute returns the Client.AuthFailure error. To get information about the block device mappings for an AMI, use the DescribeImages action.

ImageId
    The ID of the AMI.

DryRun
    Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_describe_import_image_tasks
    Describe Import Image Tasks

Description
    Displays details about an import virtual machine or import snapshot tasks that are already created.

Usage
    ec2_describe_import_image_tasks(
        DryRun = NULL,
        Filters = NULL,
        ImportTaskId = NULL,
        MaxResults = NULL,
        NextToken = NULL,
        simplify = TRUE,
        others = list(),
        print_on_error = aws_get_print_on_error(),
        retry_time = aws_get_retry_time(),
        network_timeout = aws_get_network_timeout(),
        region = aws_get_region()
    )
ec2_describe_import_image_tasks

Arguments

**DryRun**
Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]

**Filters**
List. Filter tasks using the task-state filter and one of the following values: active, completed,...[optional]

**ImportTaskId**
List. The IDs of the import image tasks.[optional]

**MaxResults**
Integer. The maximum number of results to return in a single call.[optional]

**NextToken**
Characters. The token for the next page of results[optional]

**Simplify**
Logical. Whether to simplify the result and handle nextToken in the response[optional]

**Others**
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

**PrintOnError**
Logical. Whether to show an error message when a network error occurs.

**RetryTime**
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

**NetworkTimeout**
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

**Region**
Character. The region of the AWS service.

Value

A list object or a character vector

**DryRun**
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

**Filters**
Filter tasks using the task-state filter and one of the following values: active, completed, deleting, or deleted.

**ImportTaskId**
The IDs of the import image tasks.

**MaxResults**
The maximum number of results to return in a single call.
ec2_describe_import_snapshot_tasks

Describe Import Snapshot Tasks

Description

Describes your import snapshot tasks.

Usage

```
ec2_describe_import_snapshot_tasks(
    DryRun = NULL,
    Filters = NULL,
    ImportTaskId = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action,</td>
</tr>
<tr>
<td></td>
<td>without actually making the request.</td>
</tr>
<tr>
<td>Filters</td>
<td>List. The filters.</td>
</tr>
<tr>
<td>ImportTaskId</td>
<td>List. A list of import snapshot task IDs.</td>
</tr>
<tr>
<td>MaxResults</td>
<td>Integer. The maximum number of results to return in a single call.</td>
</tr>
<tr>
<td>NextToken</td>
<td>Characters. The token for the next page of results.</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response.</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters</td>
</tr>
<tr>
<td></td>
<td>and need to be added into the request.</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network</td>
</tr>
<tr>
<td></td>
<td>issue. If the request has been sent retry_time times but still not able to</td>
</tr>
<tr>
<td></td>
<td>get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up.</td>
</tr>
<tr>
<td></td>
<td>Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>
Value
A list object or a character vector

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Filters
The filters.

ImportTaskId
A list of import snapshot task IDs.

MaxResults
The maximum number of results to return in a single call. To retrieve the remaining results, make another call with the returned NextToken value.

---

**ec2_describe_instances**

*Describe Instances*

---

**Description**

Describe Instances

**Usage**

```r
ec2_describe_instances(
  Filter = NULL,
  InstanceId = NULL,
  DryRun = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
Arguments

Filter: Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]
InstanceId: List. The instance IDs. Default: Describes all your instances. [optional]
DryRun: Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]
MaxResults: Integer. The maximum number of results to return in a single call.[optional]
NextToken: Characters. The token for the next page of results[optional]
simplify: Logical. Whether to simplify the result and handle nextToken in the response[optional]
others: Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error: Logical. Whether to show an error message when a network error occurs.
retry_time: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region: Character. The region of the AWS service.

Value

A list object or a character vector

Filter

The filters.

- affinity - The affinity setting for an instance running on a Dedicated Host (default \ host).
- architecture - The instance architecture (x86_64 | arm64).
- availability-zone - The Availability Zone of the instance.
- block-device-mapping.attach-time - The attach time for an EBS volume mapped to the instance, for example, 2010-09-15T17:15:20.000Z.
- block-device-mapping.delete-on-termination - A Boolean that indicates whether the EBS volume is deleted on instance termination.
- block-device-mapping.device-name - The device name specified in the block device mapping (for example, /dev/sdh or xvdh).
- block-device-mapping.status - The status for the EBS volume (attaching | attached | detaching | detached).
- block-device-mapping.volume-id - The volume ID of the EBS volume.
- client-token - The idempotency token you provided when you launched the instance.
- dns-name - The public DNS name of the instance.
- group-id - The ID of the security group for the instance. EC2-Classic only.
• group-name - The name of the security group for the instance. EC2-Classic only.
• hibernation-options.configured - A Boolean that indicates whether the instance is enabled for hibernation. A value of true means that the instance is enabled for hibernation.
• host-id - The ID of the Dedicated Host on which the instance is running, if applicable.
• hypervisor - The hypervisor type of the instance (ovm \| xen). The value xen is used for both Xen and Nitro hypervisors.
• iam-instance-profile.arn - The instance profile associated with the instance. Specified as an ARN.
• image-id - The ID of the image used to launch the instance.
• instance-id - The ID of the instance.
• instance-lifecycle - Indicates whether this is a Spot Instance or a Scheduled Instance (spot \| scheduled).
• instance-state-code - The state of the instance, as a 16-bit unsigned integer. The high byte is used for internal purposes and should be ignored. The low byte is set based on the state represented. The valid values are: 0 (pending), 16 (running), 32 (shutting-down), 48 (terminated), 64 (stopping), and 80 (stopped).
• instance-state-name - The state of the instance (pending \| running \| shutting-down \| terminated \| stopping \| stopped).
• instance-type - The type of instance (for example, t2.micro).
• instance.group-id - The ID of the security group for the instance.
• instance.group-name - The name of the security group for the instance.
• ip-address - The public IPv4 address of the instance.
• kernel-id - The kernel ID.
• key-name - The name of the key pair used when the instance was launched.
• launch-index - When launching multiple instances, this is the index for the instance in the launch group (for example, 0, 1, 2, and so on).
• launch-time - The time when the instance was launched.
• metadata-options.http-tokens - The metadata request authorization state (optional \| required)
• metadata-options.http-put-response-hop-limit - The http metadata request put response hop limit (integer, possible values 1 to 64)
• metadata-options.http-endpoint - Enable or disable metadata access on http endpoint (enabled \| disabled)
• monitoring-state - Indicates whether detailed monitoring is enabled (disabled \| enabled).
• network-interface.addresses.private-ip-address - The private IPv4 address associated with the network interface.
• network-interface.addresses.primary - Specifies whether the IPv4 address of the network interface is the primary private IPv4 address.
• network-interface.addresses.association.public-ip - The ID of the association of an Elastic IP address (IPv4) with a network interface.
• `network-interface.addresses.association.ip-owner-id` - The owner ID of the private IPv4 address associated with the network interface.

• `network-interface.association.public-ip` - The address of the Elastic IP address (IPv4) bound to the network interface.

• `network-interface.association.ip-owner-id` - The owner of the Elastic IP address (IPv4) associated with the network interface.

• `network-interface.association.allocation-id` - The allocation ID returned when you allocated the Elastic IP address (IPv4) for your network interface.

• `network-interface.association.association-id` - The association ID returned when the network interface was associated with an IPv4 address.

• `network-interface.attachment.attachment-id` - The ID of the interface attachment.

• `network-interface.attachment.instance-id` - The ID of the instance to which the network interface is attached.

• `network-interface.attachment.instance-owner-id` - The owner ID of the instance to which the network interface is attached.

• `network-interface.attachment.device-index` - The device index to which the network interface is attached.

• `network-interface.attachment.status` - The status of the attachment (attaching \| attached \| detaching \| detached).

• `network-interface.attachment.attach-time` - The time that the network interface was attached to an instance.

• `network-interface.attachment.delete-on-termination` - Specifies whether the attachment is deleted when an instance is terminated.

• `network-interface.availability-zone` - The Availability Zone for the network interface.

• `network-interface.description` - The description of the network interface.

• `network-interface.group-id` - The ID of a security group associated with the network interface.

• `network-interface.group-name` - The name of a security group associated with the network interface.

• `network-interface.ipv6-addresses.ipv6-address` - The IPv6 address associated with the network interface.

• `network-interface.mac-address` - The MAC address of the network interface.

• `network-interface.network-interface-id` - The ID of the network interface.

• `network-interface.owner-id` - The ID of the owner of the network interface.

• `network-interface.private-dns-name` - The private DNS name of the network interface.

• `network-interface.requester-id` - The requester ID for the network interface.

• `network-interface.requester-managed` - Indicates whether the network interface is being managed by AWS.

• `network-interface.status` - The status of the network interface (available \| in-use).

• `network-interface.source-dest-check` - Whether the network interface performs source/destination checking. A value of true means that checking is enabled, and false means that checking is disabled. The value must be false for the network interface to perform network address translation (NAT) in your VPC.
ec2_describe_instances

- network-interface.subnet-id - The ID of the subnet for the network interface.
- network-interface.vpc-id - The ID of the VPC for the network interface.
- owner-id - The AWS account ID of the instance owner.
- placement-group-name - The name of the placement group for the instance.
- placement-partition-number - The partition in which the instance is located.
- platform - The platform. To list only Windows instances, use windows.
- private-dns-name - The private IPv4 DNS name of the instance.
- private-ip-address - The private IPv4 address of the instance.
- product-code - The product code associated with the AMI used to launch the instance.
- product-code.type - The type of product code (devpay | marketplace).
- ramdisk-id - The RAM disk ID.
- reason - The reason for the current state of the instance (for example, shows \User Initiated [date]\ when you stop or terminate the instance). Similar to the state-reason-code filter.
- requester-id - The ID of the entity that launched the instance on your behalf (for example, AWS Management Console, Auto Scaling, and so on).
- reservation-id - The ID of the instance\'s reservation. A reservation ID is created any time you launch an instance. A reservation ID has a one-to-one relationship with an instance launch request, but can be associated with more than one instance if you launch multiple instances using the same launch request. For example, if you launch one instance, you get one reservation ID. If you launch ten instances using the same launch request, you also get one reservation ID.
- root-device-name - The device name of the root device volume (for example, /dev/sda1).
- root-device-type - The type of the root device volume (ebs | instance-store).
- source-dest-check - Indicates whether the instance performs source/destination checking. A value of true means that checking is enabled, and false means that checking is disabled. The value must be false for the instance to perform network address translation (NAT) in your VPC.
- spot-instance-request-id - The ID of the Spot Instance request.
- state-reason-code - The reason code for the state change.
- state-reason-message - A message that describes the state change.
- subnet-id - The ID of the subnet for the instance.
- tag:\ - The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA for the filter value.
- tag-key - The key of a tag assigned to the resource. Use this filter to find all resources that have a tag with a specific key, regardless of the tag value.
- tenancy - The tenancy of an instance (dedicated | default | host).
- virtualization-type - The virtualization type of the instance (paravirtual | hvm).
- vpc-id - The ID of the VPC that the instance is running in.
Describe Instance Attribute

Description

Describes the specified attribute of the specified instance. You can specify only one attribute at a time. Valid attribute values are:
- instanceType
- kernel
- ramdisk
- userData
- disableApiTermination
- instanceInitiatedShutdownBehavior
- rootDeviceName
- blockDeviceMapping
- productCodes
- sourceDestCheck
- groupSet
- ebsOptimized
- sriovNetSupport

Usage

```r
ec2_describe_instance_attribute(
  Attribute,
  InstanceId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **Attribute** Character. The instance attribute. Note: The enaSupport attribute is not supported at this time.
- **InstanceId** Character. The ID of the instance.
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]

others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.

Value
A list object or a character vector

Attribute
The instance attribute.

Note: The enaSupport attribute is not supported at this time.

InstanceId
The ID of the instance.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

describe_instance_credit_specifications

Describe Instance Credit Specifications

Description
Describe Instance Credit Specifications
Usage

e2_describe_instance_credit_specifications(
    DryRun = NULL,
    Filter = NULL,
    InstanceId = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
Filter Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]
InstanceId List. The instance IDs.[optional]
MaxResults Integer. The maximum number of results to return in a single call.[optional]
NextToken Characters. The token for the next page of results[optional]
simplify Logical. Whether to simplify the result and handle NextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value

A list object or a character vector

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
Filter
The filters.

• instance-id - The ID of the instance.

InstanceId
The instance IDs.
Default: Describes all your instances.
Constraints: Maximum 1000 explicitly specified instance IDs.

MaxResults
The maximum number of results to return in a single call. To retrieve the remaining results, make another call with the returned NextToken value. This value can be between 5 and 1000. You cannot specify this parameter and the instance IDs parameter in the same call.

ec2_describe_instance_event_notification_attributes
Describe Instance Event Notification Attributes

Description
Describes the tag keys that are registered to appear in scheduled event notifications for resources in the current Region.

Usage
ec2_describe_instance_event_notification_attributes(
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value
A list object or a character vector

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_describe_instance_status

Describe Instance Status

Description
Describe Instance Status

Usage

c2.describe_instance_status(
    Filter = NULL,
    InstanceId = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    DryRun = NULL,
    IncludeAllInstances = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

ec2_describe_instance_status

Arguments

Filter
Named list where the name is the filter name and the value is the filter's value.
The value can be a vector or a list object (see below)[optional]

InstanceId
List. The instance IDs.[optional]

MaxResults
Integer. The maximum number of results to return in a single call.[optional]

NextToken
Characters. The token for the next page of results[optional]

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

IncludeAllInstances
Logical. When true, includes the health status for all instances.[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
	network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value

A list object or a character vector

Filter

The filters.

- availability-zone - The Availability Zone of the instance.
- event.code - The code for the scheduled event (instance-reboot \ system-reboot \ system-maintenance \ instance-retirement \ instance-stop).
- event.description - A description of the event.
- event.instance-event-id - The ID of the event whose date and time you are modifying.
- event.not-after - The latest end time for the scheduled event (for example, 2014-09-15T17:15:20.000Z).
- event.not-before - The earliest start time for the scheduled event (for example, 2014-09-15T17:15:20.000Z).
- event.not-before-deadline - The deadline for starting the event (for example, 2014-09-15T17:15:20.000Z).
- instance-state-code - The code for the instance state, as a 16-bit unsigned integer. The high byte is used for internal purposes and should be ignored. The low byte is set based on the state represented. The valid values are 0 (pending), 16 (running), 32 (shutting-down), 48 (terminated), 64 (stopping), and 80 (stopped).
• `instance-state-name` - The state of the instance (pending | running | shutting-down | terminated | stopping | stopped).
• `instance-status.reachability` - Filters on instance status where the name is reachability (passed | failed | initializing | insufficient-data).
• `instance-status.status` - The status of the instance (ok | impaired | initializing | insufficient-data | not-applicable).
• `system-status.reachability` - Filters on system status where the name is reachability (passed | failed | initializing | insufficient-data).
• `system-status.status` - The system status of the instance (ok | impaired | initializing | insufficient-data | not-applicable).

**InstanceId**

The instance IDs.

- Default: Describes all your instances.
- Constraints: Maximum 100 explicitly specified instance IDs.

**MaxResults**

The maximum number of results to return in a single call. To retrieve the remaining results, make another call with the returned `NextToken` value. This value can be between 5 and 1000. You cannot specify this parameter and the instance IDs parameter in the same call.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

**IncludeAllInstances**

When `true`, includes the health status for all instances. When `false`, includes the health status for running instances only.

- Default: `false`

**Description**

Describes the details of the instance types that are offered in a location. The results can be filtered by the attributes of the instance types.
Usage

e2_describe_instance_types(
    DryRun = NULL,
    InstanceType = NULL,
    Filter = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request,[optional]

InstanceType List. The instance types.[optional]

Filter Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]

MaxResults Integer. The maximum number of results to return for the request in a single page.[optional]

NextToken Characters. The token for the next page of results[optional]

simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]

others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.

Value

A list object or a character vector

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
ec2_describe_instance_types

345

**InstanceType**

The instance types. For more information, see Instance types in the Amazon EC2 User Guide.

**Filter**

One or more filters. Filter names and values are case-sensitive.

- **auto-recovery-supported** - Indicates whether auto recovery is supported (true \| false).
- **bare-metal** - Indicates whether it is a bare metal instance type (true \| false).
- **burstable-performance-supported** - Indicates whether it is a burstable performance instance type (true \| false).
- **current-generation** - Indicates whether this instance type is the latest generation instance type of an instance family (true \| false).
- **ebs-info.ebs-optimized-info.baseline-bandwidth-in-mbps** - The baseline bandwidth performance for an EBS-optimized instance type, in Mbps.
- **ebs-info.ebs-optimized-info.baseline-iops** - The baseline input/output storage operations per second for an EBS-optimized instance type.
- **ebs-info.ebs-optimized-info.baseline-throughput-in-mbps** - The baseline throughput performance for an EBS-optimized instance type, in MB/s.
- **ebs-info.ebs-optimized-info.maximum-bandwidth-in-mbps** - The maximum bandwidth performance for an EBS-optimized instance type, in Mbps.
- **ebs-info.ebs-optimized-info.maximum-iops** - The maximum input/output storage operations per second for an EBS-optimized instance type.
- **ebs-info.ebs-optimized-info.maximum-throughput-in-mbps** - The maximum throughput performance for an EBS-optimized instance type, in MB/s.
- **ebs-info.ebs-optimized-support** - Indicates whether the instance type is EBS-optimized (supported \| unsupported \| default).
- **ebs-info.encryption-support** - Indicates whether EBS encryption is supported (supported \| unsupported).
- **ebs-info.nvme-support** - Indicates whether non-volatile memory express (NVMe) is supported for EBS volumes (required \| supported \| unsupported).
- **free-tier-eligible** - Indicates whether the instance type is eligible to use in the free tier (true \| false).
- **hibernation-supported** - Indicates whether On-Demand hibernation is supported (true \| false).
- **hypervisor** - The hypervisor (nitro \| xen).
- **instance-storage-info.disk.count** - The number of local disks.
- **instance-storage-info.disk.size-in-gb** - The storage size of each instance storage disk, in GB.
- **instance-storage-info.disk.type** - The storage technology for the local instance storage disks (hdd \| ssd).
- **instance-storage-info.nvme-support** - Indicates whether non-volatile memory express (NVMe) is supported for instance store (required \| supported \| unsupported).
- `instance-storage-info.total-size-in-gb` - The total amount of storage available from all local instance storage, in GB.
- `instance-storage-supported` - Indicates whether the instance type has local instance storage (true \| false).
- `instance-type` - The instance type (for example c5.2xlarge or c5\*).
- `memory-info.size-in-mib` - The memory size.
- `network-info.efa-supported` - Indicates whether the instance type supports Elastic Fabric Adapter (EFA) (true \| false).
- `network-info.ena-support` - Indicates whether Elastic Network Adapter (ENA) is supported or required (required \| supported \| unsupported).
- `network-info.ipv4-addresses-per-interface` - The maximum number of private IPv4 addresses per network interface.
- `network-info.ipv6-addresses-per-interface` - The maximum number of private IPv6 addresses per network interface.
- `network-info.ipv6-supported` - Indicates whether the instance type supports IPv6 (true \| false).
- `network-info.maximum-network-interfaces` - The maximum number of network interfaces per instance.
- `network-info.network-performance` - The network performance (for example, '25 Giga-bit\').
- `processor-info.supported-architecture` - The CPU architecture (arm64 \| i386 \| x86_64).
- `processor-info.sustained-clock-speed-in-ghz` - The CPU clock speed, in GHz.
- `supported-root-device-type` - The root device type (ebs \| instance-store).
- `supported-usage-class` - The usage class (on-demand \| spot).
- `supported-virtualization-type` - The virtualization type (hvm \| paravirtual).
- `vcpu-info.default-cores` - The default number of cores for the instance type.
- `vcpu-info.default-threads-per-core` - The default number of threads per core for the instance type.
- `vcpu-info.default-vcpus` - The default number of vCPUs for the instance type.
- `vcpu-info.valid-cores` - The number of cores that can be configured for the instance type.
- `vcpu-info.valid-threads-per-core` - The number of threads per core that can be configured for the instance type. For example, '1' \| '1,2'.

**MaxResults**

The maximum number of results to return for the request in a single page. The remaining results can be seen by sending another request with the next token value.
Description

Returns a list of all instance types offered. The results can be filtered by location (Region or Availability Zone). If no location is specified, the instance types offered in the current Region are returned.

Usage

c2_describe_instance_type_offerings(  
  DryRun = NULL,  
  LocationType = NULL,  
  Filter = NULL,  
  MaxResults = NULL,  
  NextToken = NULL,  
  simplify = TRUE,  
  others = list(),  
  print_on_error = aws_get_print_on_error(),  
  retry_time = aws_get_retry_time(),  
  network_timeout = aws_get_network_timeout(),  
  region = aws_get_region()  
)

Arguments

DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

LocationType Character. The location type.[optional]

Filter Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]

MaxResults Integer. The maximum number of results to return for the request in a single page.[optional]

NextToken Characters. The token for the next page of results[optional]

simplify Logical. Whether to simplify the result and handle NextToken in the response[optional]

others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
  Character. The region of the AWS service.

Value
  A list object or a character vector

DryRun
  Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

LocationType
  The location type.

Filter
  One or more filters. Filter names and values are case-sensitive.
  - location - This depends on the location type. For example, if the location type is region (default), the location is the Region code (for example, us-east-2.)
  - instance-type - The instance type. For example, c5.2xlarge.

MaxResults
  The maximum number of results to return for the request in a single page. The remaining results can be seen by sending another request with the next token value.

---

**ec2_describe_internet_gateways**

*Describe Internet Gateways*

---

**Description**

Describes one or more of your internet gateways.

**Usage**

```r
ec2_describe_internet_gateways(
  Filter = NULL,
  DryRun = NULL,
  InternetGatewayId = NULL,
  NextToken = NULL,
  MaxResults = NULL,
```
simplify = TRUE,
others = list(),
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

Filter: Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below) [optional]

DryRun: Logical. Checks whether you have the required permissions for the action, without actually making the request... [optional]

InternetGatewayId: List. One or more internet gateway IDs. Default: Describes all your internet gateways. [optional]

NextToken: Characters. The token for the next page of results [optional]

MaxResults: Integer. The maximum number of results to return with a single call. [optional]

simplify: Logical. Whether to simplify the result and handle nextToken in the response [optional]

others: Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]

print_on_error: Logical. Whether to show an error message when a network error occurs.

retry_time: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region: Character. The region of the AWS service.

Value

A list object or a character vector

Filter

One or more filters.

- attachment.state - The current state of the attachment between the gateway and the VPC (available). Present only if a VPC is attached.
- attachment.vpc-id - The ID of an attached VPC.
- internet-gateway-id - The ID of the Internet gateway.
- owner-id - The ID of the AWS account that owns the internet gateway.
• tag:\ - The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA for the filter value.

• tag-key - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

**InternetGatewayId**

One or more internet gateway IDs.

Default: Describes all your internet gateways.

**MaxResults**

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

dry_run

```python
c2.describe_ipv6_pools
```

### Description

Describes your IPv6 address pools.

### Usage

```python
c2.describe_ipv6_pools(
    PoolId = NULL,
    NextToken = NULL,
    MaxResults = NULL,
    DryRun = NULL,
    Filter = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```
Arguments

- **PoolId**: List. The IDs of the IPv6 address pools. [optional]
- **NextToken**: Characters. The token for the next page of results. [optional]
- **MaxResults**: Integer. The maximum number of results to return with a single call. [optional]
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **Filter**: Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below). [optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

**PoolId**

The IDs of the IPv6 address pools.

**MaxResults**

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned `nextToken` value.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

**Filter**

One or more filters.

- **tag**: The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key `Owner` and the value `TeamA`, specify `tag:Owner` for the filter name and `TeamA` for the filter value.
- tag-key - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.

### ec2_describe_key_pairs

**Describe Key Pairs**

#### Description

Describe Key Pairs

#### Usage

```r
ec2_describe_key_pairs(
    Filter = NULL,
    KeyName = NULL,
    KeyPairId = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

#### Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filter</td>
<td>Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]</td>
</tr>
<tr>
<td>KeyName</td>
<td>List. The key pair names. Default: Describes all your key pairs. [optional]</td>
</tr>
<tr>
<td>KeyPairId</td>
<td>List. The IDs of the key pairs,[optional]</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request,[optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response[optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>
Value

A list object or a character vector

Filter

The filters.

- **key-pair-id** - The ID of the key pair.
- **fingerprint** - The fingerprint of the key pair.
- **key-name** - The name of the key pair.
- **tag-key** - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.
- **tag:** - The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key `Owner` and the value `TeamA`, specify `tag:Owner` for the filter name and `TeamA` for the filter value.

KeyName

The key pair names.

Default: Describes all your key pairs.

KeyPairId

The IDs of the key pairs.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`. 

Description

Describes one or more launch templates.
Usage

```
ec2_describe_launch_templates(
  DryRun = NULL,
  LaunchTemplateId = NULL,
  LaunchTemplateName = NULL,
  Filter = NULL,
  NextToken = NULL,
  MaxResults = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **DryRun** Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **LaunchTemplateId** List. One or more launch template IDs. [optional]
- **LaunchTemplateName** List. One or more launch template names. [optional]
- **Filter** Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below). [optional]
- **NextToken** Characters. The token for the next page of results. [optional]
- **MaxResults** Integer. The maximum number of results to return in a single call. [optional]
- **simplify** Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]
- **others** Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]
- **print_on_error** Logical. Whether to show an error message when a network error occurs.
- **retry_time** Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout** Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region** Character. The region of the AWS service.

Value

A list object or a character vector
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

LaunchTemplateId

One or more launch template IDs.

LaunchTemplateName

One or more launch template names.

Filter

One or more filters.

- create-time - The time the launch template was created.
- launch-template-name - The name of the launch template.
- tag: - The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA for the filter value.
- tag-key - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.

MaxResults

The maximum number of results to return in a single call. To retrieve the remaining results, make another call with the returned NextToken value. This value can be between 1 and 200.
Use

ec2_describe_launch_template_versions(
    DryRun = NULL,
    LaunchTemplateId = NULL,
    LaunchTemplateName = NULL,
    LaunchTemplateVersion = NULL,
    MinVersion = NULL,
    MaxVersion = NULL,
    NextToken = NULL,
    MaxResults = NULL,
    Filter = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request,[optional]
LaunchTemplateId Character. The ID of the launch template.[optional]
LaunchTemplateName Character. The name of the launch template.[optional]
LaunchTemplateVersion List. One or more versions of the launch template.[optional]
MinVersion Character. The version number after which to describe launch template versions.[optional]
MaxVersion Character. The version number up to which to describe launch template versions.[optional]
NextToken Characters. The token for the next page of results[optional]
MaxResults Integer. The maximum number of results to return in a single call.[optional]
Filter Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout

Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region

Character. The region of the AWS service.

Value

A list object or a character vector

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

LaunchTemplateId

The ID of the launch template. To describe one or more versions of a specified launch template, you must specify either the launch template ID or the launch template name in the request. To describe all the latest or default launch template versions in your account, you must omit this parameter.

LaunchTemplateName

The name of the launch template. To describe one or more versions of a specified launch template, you must specify either the launch template ID or the launch template name in the request. To describe all the latest or default launch template versions in your account, you must omit this parameter.

LaunchTemplateVersion

One or more versions of the launch template. Valid values depend on whether you are describing a specified launch template (by ID or name) or all launch templates in your account.

To describe one or more versions of a specified launch template, valid values are $Latest, $Default, and numbers.

To describe all launch templates in your account that are defined as the latest version, the valid value is $Latest. To describe all launch templates in your account that are defined as the default version, the valid value is $Default. You can specify $Latest and $Default in the same call. You cannot specify numbers.

MinVersion

The version number after which to describe launch template versions.

MaxVersion

The version number up to which to describe launch template versions.
MaxResults

The maximum number of results to return in a single call. To retrieve the remaining results, make another call with the returned NextToken value. This value can be between 1 and 200.

Filter

One or more filters.

- create-time - The time the launch template version was created.
- ebs-optimized - A boolean that indicates whether the instance is optimized for Amazon EBS I/O.
- iam-instance-profile - The ARN of the IAM instance profile.
- image-id - The ID of the AMI.
- instance-type - The instance type.
- is-default-version - A boolean that indicates whether the launch template version is the default version.
- kernel-id - The kernel ID.
- ram-disk-id - The RAM disk ID.

---

**ec2_describe_local_gateways**

*Describe Local Gateways*

---

**Description**

Describes one or more local gateways. By default, all local gateways are described. Alternatively, you can filter the results.

**Usage**

```r
ec2_describe_local_gateways(
  LocalGatewayId = NULL,
  Filter = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
Arguments

- **LocalGatewayId** List. One or more filters.[optional]
- **Filter** Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below).[optional]
- **MaxResults** Integer. The maximum number of results to return with a single call.[optional]
- **NextToken** Characters. The token for the next page of results[optional]
- **DryRun** Logical. Checks whether you have the required permissions for the action, without actually making the request,[optional]
- **simplify** Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others** Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error** Logical. Whether to show an error message when a network error occurs.
- **retry_time** Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- **network_timeout** Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region** Character. The region of the AWS service.

Value

A list object or a character vector

**LocalGatewayId**

One or more filters.

- local-gateway-id - The ID of a local gateway.
- local-gateway-route-table-id - The ID of the local gateway route table.
- local-gateway-route-table-virtual-interface-group-association-id - The ID of the association.
- local-gateway-route-table-virtual-interface-group-id - The ID of the virtual interface group.
- outpost-arn - The Amazon Resource Name (ARN) of the Outpost.
- state - The state of the association.

**Filter**

One or more filters.

**MaxResults**

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_describe_local_gateway_route_tables

Describe Local Gateway Route Tables

Description

Describes one or more local gateway route tables. By default, all local gateway route tables are described. Alternatively, you can filter the results.

Usage

```r
ec2_describe_local_gateway_route_tables(
  LocalGatewayRouteTableId = NULL,
  Filter = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **LocalGatewayRouteTableId**
  - List. The IDs of the local gateway route tables.[optional]
- **Filter**
  - Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]
- **MaxResults**
  - Integer. The maximum number of results to return with a single call.[optional]
- **NextToken**
  - Characters. The token for the next page of results[optional]
- **DryRun**
  - Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
- **simplify**
  - Logical. Whether to simplify the result and handle `nextToken` in the response[optional]
- **others**
  - Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**
  - Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  Character. The region of the AWS service.

Value
A list object or a character vector

LocalGatewayRouteTableId
The IDs of the local gateway route tables.

Filter
One or more filters.

• local-gateway-id - The ID of a local gateway.
• local-gateway-route-table-id - The ID of a local gateway route table.
• outpost-arn - The Amazon Resource Name (ARN) of the Outpost.
• state - The state of the local gateway route table.

MaxResults
The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description
Describes the associations between virtual interface groups and local gateway route tables.
Usage

```r
ec2_describe_local_gateway_route_table_virtual_interface_group_associations(
  LocalGatewayRouteTableVirtualInterfaceGroupAssociationId = NULL,
  Filter = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- **LocalGatewayRouteTableVirtualInterfaceGroupAssociationId**
  List. The IDs of the associations. [optional]

- **Filter**
  Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below). [optional]

- **MaxResults**
  Integer. The maximum number of results to return with a single call. [optional]

- **NextToken**
  Characters. The token for the next page of results. [optional]

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]

- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  Character. The region of the AWS service.

**Value**

A list object or a character vector

**LocalGatewayRouteTableVirtualInterfaceGroupAssociationId**

The IDs of the associations.
Filter

One or more filters.

- `local-gateway-id` - The ID of a local gateway.
- `local-gateway-route-table-id` - The ID of the local gateway route table.
- `local-gateway-route-table-virtual-interface-group-association-id` - The ID of the association.
- `local-gateway-route-table-virtual-interface-group-id` - The ID of the virtual interface group.
- `state` - The state of the association.

MaxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned `nextToken` value.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

---

describe_local_gateway_route_table_vpc_associations

Describe Local Gateway Route Table Vpc Associations

Description

Describes the specified associations between VPCs and local gateway route tables.

Usage

describe_local_gateway_route_table_vpcAssociations(
    LocalGatewayRouteTableVpcAssociationId = NULL,
    Filter = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
Arguments

LocalGatewayRouteTableVpcAssociationId
- List. The IDs of the associations.[optional]

Filter
- Named list where the name is the filter name and the value is the filter's value.
  The value can be a vector or a list object (see below).[optional]

MaxResults
- Integer. The maximum number of results to return with a single call.[optional]

NextToken
- Characters. The token for the next page of results[optional]

DryRun
- Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]

simplify
- Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
- Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
- Logical. Whether to show an error message when a network error occurs.

retry_time
- Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
- Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
- Character. The region of the AWS service.

Value

- A list object or a character vector

LocalGatewayRouteTableVpcAssociationId
- The IDs of the associations.

Filter

- One or more filters.
  - local-gateway-id - The ID of a local gateway.
  - local-gateway-route-table-id - The ID of the local gateway route table.
  - local-gateway-route-table-vpc-association-id - The ID of the association.
  - state - The state of the association.
  - vpc-id - The ID of the VPC.

MaxResults

- The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description

Describes the specified local gateway virtual interfaces.

Usage

```r
ec2_describe_local_gateway_virtual_interfaces(
    LocalGatewayVirtualInterfaceId = NULL,
    Filter = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **LocalGatewayVirtualInterfaceId**
  - List. The IDs of the virtual interfaces.[optional]
- **Filter**
  - Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]
- **MaxResults**
  - Integer. The maximum number of results to return with a single call.[optional]
- **NextToken**
  - Characters. The token for the next page of results[optional]
- **DryRun**
  - Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
- **simplify**
  - Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others**
  - Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**
  - Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  Character. The region of the AWS service.

Value
A list object or a character vector

LocalGatewayVirtualInterfaceId
The IDs of the virtual interfaces.

Filter
One or more filters.

MaxResults
The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

ec2_describe_local_gateway_virtual_interface_groups

Describe Local Gateway Virtual Interface Groups

Description
Describes the specified local gateway virtual interface groups.

Usage

ec2_describe_local_gateway_virtual_interface_groups(
    LocalGatewayVirtualInterfaceGroupId = NULL,
    Filter = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    DryRun = NULL,
)
simplify = TRUE,
others = list(),
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

LocalGatewayVirtualInterfaceGroupId
  List. The IDs of the virtual interface groups.[optional]

Filter
  Named list where the name is the filter name and the value is the filter’s value.
  The value can be a vector or a list object (see below)[optional]

MaxResults
  Integer. The maximum number of results to return with a single call.[optional]

NextToken
  Characters. The token for the next page of results[optional]

DryRun
  Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

simplify
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
  Logical. Whether to show an error message when a network error occurs.

retry_time
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
  Character. The region of the AWS service.

Value

A list object or a character vector

LocalGatewayVirtualInterfaceGroupId
  The IDs of the virtual interface groups.

Filter

One or more filters.

  • local-gateway-id - The ID of a local gateway.
  • local-gateway-virtual-interface-id - The ID of the virtual interface.
  • local-gateway-virtual-interface-group-id - The ID of the virtual interface group.
MaxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

describe_managed_prefix_lists

Describe Managed Prefix Lists

Description

Describe Managed Prefix Lists

Usage

describe_managed_prefix_lists(
    DryRun = NULL,
    Filter = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    PrefixListId = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

Filter Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]

MaxResults Integer. The maximum number of results to return with a single call.[optional]

NextToken Characters. The token for the next page of results[optional]

PrefixListId List. One or more prefix list IDs,[optional]

simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  Character. The region of the AWS service.

Value

A list object or a character vector

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Filter

One or more filters.

- owner-id - The ID of the prefix list owner.
- prefix-list-id - The ID of the prefix list.
- prefix-list-name - The name of the prefix list.

MaxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

PrefixListId

One or more prefix list IDs.
ec2_describe_moving_addresses

Description

Describes your Elastic IP addresses that are being moved to the EC2-VPC platform, or that are being restored to the EC2-Classic platform. This request does not return information about any other Elastic IP addresses in your account.

Usage

ec2_describe_moving_addresses(
    Filter = NULL,
    DryRun = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    PublicIp = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

Filter               Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]
DryRun               Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
MaxResults           Integer. The maximum number of results to return for the request in a single page.[optional]
NextToken            Characters. The token for the next page of results[optional]
PublicIp             List. One or more Elastic IP addresses.[optional]
simplify             Logical. Whether to simplify the result and handle nextToken in the response[optional]
others               Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error       Logical. Whether to show an error message when a network error occurs.
retry_time           Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not able to get the response, an error will be thrown.
network_timeout
   Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
   Character. The region of the AWS service.

Value
   A list object or a character vector

Filter
   One or more filters.
      • moving-status - The status of the Elastic IP address (MovingToVpc | RestoringToClassic).

DryRun
   Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

MaxResults
   The maximum number of results to return for the request in a single page. The remaining results of the initial request can be seen by sending another request with the returned NextToken value. This value can be between 5 and 1000; if MaxResults is given a value outside of this range, an error is returned.
   Default: If no value is provided, the default is 1000.

PublicIp
   One or more Elastic IP addresses.

ec2_describe_nat_gateways

Describe Nat Gateways

Description
   Describes one or more of your NAT gateways.
Usage

```r
ec2_describe_nat_gateways(
  DryRun = NULL,
  Filter = NULL,
  MaxResults = NULL,
  NatGatewayId = NULL,
  NextToken = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]

- **Filter**
  Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]

- **MaxResults**
  Integer. The maximum number of results to return with a single call.[optional]

- **NatGatewayId**
  List. One or more NAT gateway IDs.[optional]

- **NextToken**
  Characters. The token for the next page of results[optional]

- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  Character. The region of the AWS service.

Value

A list object or a character vector

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
Filter

One or more filters.

- nat-gateway-id - The ID of the NAT gateway.
- state - The state of the NAT gateway (pending | failed | available | deleting | deleted).
- subnet-id - The ID of the subnet in which the NAT gateway resides.
- tag:
  - The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA for the filter value.
- tag-key - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.
- vpc-id - The ID of the VPC in which the NAT gateway resides.

MaxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

NatGatewayId

One or more NAT gateway IDs.

Description

Describe Network Acls

Usage

```r
ec2_describe_network_acls(
    Filter = NULL,
    DryRun = NULL,
    NetworkAclId = NULL,
    NextToken = NULL,
    MaxResults = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```
Arguments

Filter  Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]

DryRun  Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

NetworkAclId  List. One or more network ACL IDs. Default: Describes all your network ACLs. [optional]

NextToken  Characters. The token for the next page of results[optional]

MaxResults  Integer. The maximum number of results to return with a single call.[optional]

simplify  Logical. Whether to simplify the result and handle nextToken in the response[optional]

others  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error  Logical. Whether to show an error message when a network error occurs.

retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  Character. The region of the AWS service.

Value

A list object or a character vector

Filter

One or more filters.

- association.association-id - The ID of an association ID for the ACL.
- association.network-acl-id - The ID of the network ACL involved in the association.
- association.subnet-id - The ID of the subnet involved in the association.
- default - Indicates whether the ACL is the default network ACL for the VPC.
- entry.cidr - The IPv4 CIDR range specified in the entry.
- entry.icmp.code - The ICMP code specified in the entry, if any.
- entry.icmp.type - The ICMP type specified in the entry, if any.
- entry.ipv6-cidr - The IPv6 CIDR range specified in the entry.
- entry.port-range.from - The start of the port range specified in the entry.
- entry.port-range.to - The end of the port range specified in the entry.
- entry.protocol - The protocol specified in the entry (tcp \ udp \ icmp or a protocol number).
- entry.rule-action - Allows or denies the matching traffic (allow \ deny).
• entry.rule-number - The number of an entry (in other words, rule) in the set of ACL entries.
• network-acl-id - The ID of the network ACL.
• owner-id - The ID of the AWS account that owns the network ACL.
• tag: - The key/value combination of a tag assigned to the resource. Use the tag key in the
  filter name and the tag value as the filter value. For example, to find all resources that have a
tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA
  for the filter value.
• tag-key - The key of a tag assigned to the resource. Use this filter to find all resources
  assigned a tag with a specific key, regardless of the tag value.
• vpc-id - The ID of the VPC for the network ACL.

DryRun

Checks whether you have the required permissions for the action, without actually making the
request, and provides an error response. If you have the required permissions, the error response is
DryRunOperation. Otherwise, it is UnauthorizedOperation.

NetworkAclId

One or more network ACL IDs.
Default: Describes all your network ACLs.

MaxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make
another call with the returned nextToken value.

ec2_describe_network_insights_analyses

Describe Network Insights Analyses

Description

Describes one or more of your network insights analyses.

Usage

ec2_describe_network_insights_analyses(
    NetworkInsightsAnalysisId = NULL,
    NetworkInsightsPathId = NULL,
    AnalysisStartTime = NULL,
    AnalysisEndTime = NULL,
    Filter = NULL,
    MaxResults = NULL,
    DryRun = NULL,
    NextToken = NULL,
simplify = TRUE,
others = list(),
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

NetworkInsightsAnalysisId
List. The ID of the network insights analyses. You must specify either analysis IDs or a path ID.[optional]

NetworkInsightsPathId
Character. The ID of the path. You must specify either a path ID or analysis IDs.[optional]

AnalysisStartTime
Character. The time when the network insights analyses started.[optional]

AnalysisEndTime
Character. The time when the network insights analyses ended.[optional]

Filter
Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]

MaxResults
Integer. The maximum number of results to return with a single call.[optional]

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

NextToken
Characters. The token for the next page of results[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value
A list object or a character vector

NetworkInsightsAnalysisId
The ID of the network insights analyses. You must specify either analysis IDs or a path ID.
NetworkInsightsPathId

The ID of the path. You must specify either a path ID or analysis IDs.

AnalysisStartTime

The time when the network insights analyses started.

AnalysisEndTime

The time when the network insights analyses ended.

Filter

The filters. The following are possible values:

- PathFound - A Boolean value that indicates whether a feasible path is found.
- Status - The status of the analysis (running \ succeeded \ failed).

MaxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

description

Describe Network Insights Paths

Description

Describes one or more of your paths.

Usage

e2_describe_network_insights_paths(
    NetworkInsightsPathId = NULL,
    Filter = NULL,
    MaxResults = NULL,
    DryRun = NULL,
    NextToken = NULL,
    simplify = TRUE,
    others = list(),
)
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

NetworkInsightsPathId
List. The IDs of the paths.[optional]

Filter
Named list where the name is the filter name and the value is the filter’s value.
The value can be a vector or a list object (see below)[optional]

MaxResults
Integer. The maximum number of results to return with a single call.[optional]

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]

NextToken
Characters. The token for the next page of results[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value
A list object or a character vector

NetworkInsightsPathId
The IDs of the paths.

Filter
The filters. The following are possible values:

- Destination - The ID of the resource.
- DestinationPort - The destination port.
- Name - The path name.
- Protocol - The protocol.
- Source - The ID of the resource.
MaxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned `nextToken` value.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

Usage

```r
ec2_describe_network_interfaces(
  Filter = NULL,
  DryRun = NULL,
  NetworkInterfaceId = NULL,
  NextToken = NULL,
  MaxResults = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **Filter**: Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
- **NetworkInterfaceId**: List. One or more network interface IDs. Default: Describes all your network interfaces. [optional]
- **NextToken**: Characters. The token for the next page of results[optional]
- **MaxResults**: Integer. The maximum number of items to return for this request.[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value
A list object or a character vector

Filter
One or more filters.

• addresses.private-ip-address - The private IPv4 addresses associated with the network interface.
• addresses.primary - Whether the private IPv4 address is the primary IP address associated with the network interface.
• addresses.association.public-ip - The association ID returned when the network interface was associated with the Elastic IP address (IPv4).
• addresses.association.owner-id - The owner ID of the addresses associated with the network interface.
• association.association-id - The association ID returned when the network interface was associated with an IPv4 address.
• association.allocation-id - The allocation ID returned when you allocated the Elastic IP address (IPv4) for your network interface.
• association.ip-owner-id - The owner of the Elastic IP address (IPv4) associated with the network interface.
• association.public-ip - The address of the Elastic IP address (IPv4) bound to the network interface.
• association.public-dns-name - The public DNS name for the network interface (IPv4).
• attachment.attachment-id - The ID of the interface attachment.
• attachment.attach-time - The time that the network interface was attached to an instance.
• attachment.delete-on-termination - Indicates whether the attachment is deleted when an instance is terminated.
• attachment.device-index - The device index to which the network interface is attached.
• attachment.instance-id - The ID of the instance to which the network interface is attached.
ec2_describe_network_interfaces

• attachment.instance-owner-id - The owner ID of the instance to which the network interface is attached.
• attachment.status - The status of the attachment (attaching | attached | detaching | detached).
• availability-zone - The Availability Zone of the network interface.
• description - The description of the network interface.
• group-id - The ID of a security group associated with the network interface.
• group-name - The name of a security group associated with the network interface.
• ipv6-addresses.ipv6-address - An IPv6 address associated with the network interface.
• mac-address - The MAC address of the network interface.
• network-interface-id - The ID of the network interface.
• owner-id - The AWS account ID of the network interface owner.
• private-ip-address - The private IPv4 address or addresses of the network interface.
• private-dns-name - The private DNS name of the network interface (IPv4).
• requester-id - The alias or AWS account ID of the principal or service that created the network interface.
• requester-managed - Indicates whether the network interface is being managed by an AWS service (for example, AWS Management Console, Auto Scaling, and so on).
• source-dest-check - Indicates whether the network interface performs source/destination checking. A value of true means checking is enabled, and false means checking is disabled. The value must be false for the network interface to perform network address translation (NAT) in your VPC.
• status - The status of the network interface. If the network interface is not attached to an instance, the status is available; if a network interface is attached to an instance the status is in-use.
• subnet-id - The ID of the subnet for the network interface.
• tag: - The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA for the filter value.
• tag-key - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.
• vpc-id - The ID of the VPC for the network interface.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

NetworkInterfaceId

One or more network interface IDs.
Default: Describes all your network interfaces.
MaxResults

The maximum number of items to return for this request. The request returns a token that you can specify in a subsequent call to get the next set of results. You cannot specify this parameter and the network interface IDs parameter in the same request.

ec2_describe_network_interface_attribute

Describe Network Interface Attribute

Description

Describes a network interface attribute. You can specify only one attribute at a time.

Usage

ec2_describe_network_interface_attribute(
    NetworkInterfaceId,
    Attribute = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

NetworkInterfaceId 
Character. The ID of the network interface.

Attribute 
Character. The attribute of the network interface. This parameter is required.[optional]

DryRun 
Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]

simplify 
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others 
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error 
Logical. Whether to show an error message when a network error occurs.

retry_time 
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout 
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region 
Character. The region of the AWS service.
Value

A list object or a character vector

NetworkInterfaceId

The ID of the network interface.

Attribute

The attribute of the network interface. This parameter is required.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**Description**

Describes the permissions for your network interfaces.

**Usage**

```r
ec2_describe_network_interface_permissions(
    NetworkInterfacePermissionId = NULL,
    Filter = NULL,
    NextToken = NULL,
    MaxResults = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

**Arguments**

- **NetworkInterfacePermissionId**
  - List. One or more network interface permission IDs.[optional]

- **Filter**
  - Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]

- **NextToken**
  - Characters. The token for the next page of results[optional]
MaxResults  Integer. The maximum number of results to return in a single call.[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value
A list object or a character vector

NetworkInterfacePermissionId
One or more network interface permission IDs.

Filter
One or more filters.

- network-interface-permission.network-interface-permission-id - The ID of the permission.
- network-interface-permission.network-interface-id - The ID of the network interface.
- network-interface-permission.aws-account-id - The AWS account ID.
- network-interface-permission.aws-service - The AWS service.
- network-interface-permission.permission - The type of permission (INSTANCE-ATTACH | EIP-ASSOCIATE).

MaxResults
The maximum number of results to return in a single call. To retrieve the remaining results, make another call with the returned NextToken value. If this parameter is not specified, up to 50 results are returned by default.
Describe Placement Groups

Description

Describes the specified placement groups or all of your placement groups. For more information, see Placement groups in the Amazon EC2 User Guide.

Usage

ec2_describe_placement_groups(
    Filter = NULL,
    DryRun = NULL,
    GroupName = NULL,
    GroupId = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

Filter Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]
GroupName List. The names of the placement groups.[optional]
GroupId List. The IDs of the placement groups.[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.
Value

A list object or a character vector

Filter

The filters.

- **group-name** - The name of the placement group.
- **state** - The state of the placement group (pending | available | deleting | deleted).
- **strategy** - The strategy of the placement group (cluster | spread | partition).
- **tag**: - The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key `Owner` and the value `TeamA`, specify `tag:Owner` for the filter name and `TeamA` for the filter value.
- **tag-key** - The key of a tag assigned to the resource. Use this filter to find all resources that have a tag with a specific key, regardless of the tag value.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

GroupName

The names of the placement groups.

Default: Describes all your placement groups, or only those otherwise specified.

GroupId

The IDs of the placement groups.

---

**ec2_describe_prefix_lists**

*Describe Prefix Lists*

**Description**

Describe Prefix Lists
Usage

```r
e2_describe_prefix_lists(
  DryRun = NULL,
  Filter = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  PrefixListId = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **Filter**: Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below) [optional]
- **MaxResults**: Integer. The maximum number of results to return with a single call [optional]
- **NextToken**: Characters. The token for the next page of results [optional]
- **PrefixListId**: List. One or more prefix list IDs [optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response [optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.
Filter

One or more filters.

- prefix-list-id: The ID of a prefix list.
- prefix-list-name: The name of a prefix list.

MaxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

PrefixListId

One or more prefix list IDs.

describe_principal_id_format

Describe Principal Id Format

Description

Describe Principal Id Format

Usage

describe_principal_id_format(
    DryRun = NULL,
    Resource = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]

Resource List. The type of resource: bundle \ conversion-task \ customer-gateway \ dhcp-options \ elastic-ip-allocation...[optional]

MaxResults Integer. The maximum number of results to return in a single call.[optional]

NextToken Characters. The token for the next page of results[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]

others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.

Value
A list object or a character vector

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Resource
The type of resource: bundle \ conversion-task \ customer-gateway \ dhcp-options \ elastic-ip-allocation \ elastic-ip-association \ export-task \ flow-log \ image \ import-task \ instance \ internet-gateway \ network-acl \ network-acl-association \ network-interface \ network-interface-attachment \ prefix-list \ reservation \ route-table \ route-table-association \ security-group \ snapshot \ subnet \ subnet-cidr-block-association \ volume \ vpc \ vpc-cidr-block-association \ vpc-endpoint \ vpc-peering-connection \ vpn-connection \ vpn-gateway

MaxResults
The maximum number of results to return in a single call. To retrieve the remaining results, make another call with the returned NextToken value.

ec2_describe_public_ipv4_pools

Describe Public Ipv4 Pools

Description
Describes the specified IPv4 address pools.
Usage

```r
ec2_describe_public_ipv4_pools(
  PoolId = NULL,
  NextToken = NULL,
  MaxResults = NULL,
  Filter = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **PoolId**
  List. The IDs of the address pools.[optional]
- **NextToken**
  Characters. The token for the next page of results.[optional]
- **MaxResults**
  Integer. The maximum number of results to return with a single call.[optional]
- **Filter**
  Named list where the name is the filter name and the value is the filter's value.
  The value can be a vector or a list object (see below).[optional]
- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response.[optional]
- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request.[optional]
- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.
- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**
  Character. The region of the AWS service.

Value

A list object or a character vector

**PoolId**

The IDs of the address pools.

**MaxResults**

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned `nextToken` value.
Filter

One or more filters.

- **tag:** - The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key `Owner` and the value `TeamA`, specify `tag:Owner` for the filter name and `TeamA` for the filter value.
- **tag-key** - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.

---

**ec2_describe_regions**  
*Describe Regions*

---

Description

Describe Regions

Usage

```r
ec2_describe_regions(
  Filter = NULL,
  RegionName = NULL,
  DryRun = NULL,
  AllRegions = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **Filter**  
  Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]
- **RegionName**  
  List. The names of the Regions.[optional]
- **DryRun**  
  Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]
- **AllRegions**  
  Logical. Indicates whether to display all Regions, including Regions that are disabled for your account.[optional]
- **simplify**  
  Logical. Whether to simplify the result and handle `nextToken` in the response[optional]
- **others**  
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**  
  Logical. Whether to show an error message when a network error occurs.
**retry_time**  
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

**network_timeout**  
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

**region**  
Character. The region of the AWS service.

**Value**
A list object or a character vector

**Filter**
The filters.

- **endpoint** - The endpoint of the Region (for example, ec2.us-east-1.amazonaws.com).
- **opt-in-status** - The opt-in status of the Region (opt-in-not-required \| opted-in \| not-opted-in).
- **region-name** - The name of the Region (for example, us-east-1).

**RegionName**
The names of the Regions. You can specify any Regions, whether they are enabled and disabled for your account.

**DryRun**
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

**AllRegions**
Indicates whether to display all Regions, including Regions that are disabled for your account.

---

**Description**
Describe Reserved Instances
Usage

```r
ec2_describe_reserved_instances(
  Filter = NULL,
  OfferingClass = NULL,
  ReservedInstancesId = NULL,
  DryRun = NULL,
  OfferingType = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **Filter**: Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]
- **OfferingClass**: Character. Describes whether the Reserved Instance is Standard or Convertible.[optional]
- **ReservedInstancesId**: List. One or more Reserved Instance IDs.[optional]
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
- **OfferingType**: Character. The Reserved Instance offering type.[optional]
- **simplify**: Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

Filter

One or more filters.
• availability-zone - The Availability Zone where the Reserved Instance can be used.
• duration - The duration of the Reserved Instance (one year or three years), in seconds (31536000 | 94608000).
• end - The time when the Reserved Instance expires (for example, 2015-08-07T11:54:42.000Z).
• fixed-price - The purchase price of the Reserved Instance (for example, 9800.0).
• instance-type - The instance type that is covered by the reservation.
• scope - The scope of the Reserved Instance (Region or Availability Zone).
• product-description - The Reserved Instance product platform description. Instances that include (Amazon VPC) in the product platform description will only be displayed to EC2-Classic account holders and are for use with Amazon VPC (Linux/UNIX | Linux/UNIX (Amazon VPC) | SUSE Linux | SUSE Linux (Amazon VPC) | Red Hat Enterprise Linux | Red Hat Enterprise Linux (Amazon VPC) | Windows | Windows (Amazon VPC) | Windows with SQL Server Standard | Windows with SQL Server Standard | Windows with SQL Server Web | Windows with SQL Server Web (Amazon VPC) | Windows with SQL Server Enterprise | Windows with SQL Server Enterprise (Amazon VPC)).
• reserved-instances-id - The ID of the Reserved Instance.
• start - The time at which the Reserved Instance purchase request was placed (for example, 2014-08-07T11:54:42.000Z).
• state - The state of the Reserved Instance (payment-pending | active | payment-failed | retired).
• tag: - The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA for the filter value.
• tag-key - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.
• usage-price - The usage price of the Reserved Instance, per hour (for example, 0.84).

OfferingClass
Describes whether the Reserved Instance is Standard or Convertible.

ReservedInstancesId
One or more Reserved Instance IDs.
Default: Describes all your Reserved Instances, or only those otherwise specified.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

OfferingType
The Reserved Instance offering type. If you are using tools that predate the 2011-11-01 API version, you only have access to the Medium Utilization Reserved Instance offering type.
ec2_describe_reserved_instances_listings

Description

Describe Reserved Instances Listings

Usage

```r
ec2_describe_reserved_instances_listings(
  Filter = NULL,
  ReservedInstancesId = NULL,
  ReservedInstancesListingId = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **Filter**: Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]
- **ReservedInstancesId**: Character. One or more Reserved Instance IDs.[optional]
- **ReservedInstancesListingId**: Character. One or more Reserved Instance listing IDs.[optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response[optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector
Filter

One or more filters.

- **reserved-instances-id** - The ID of the Reserved Instances.
- **reserved-instances-listing-id** - The ID of the Reserved Instances listing.
- **status** - The status of the Reserved Instance listing (pending | active | cancelled | closed).
- **status-message** - The reason for the status.

ReservedInstancesId

One or more Reserved Instance IDs.

ReservedInstancesListingId

One or more Reserved Instance listing IDs.

---

**ec2_describe_reserved_instances_modifications**

*Describe Reserved Instances Modifications*

---

Description

Describe Reserved Instances Modifications

Usage

```r
ec2_describe_reserved_instances_modifications(
  Filter = NULL,
  ReservedInstancesModificationId = NULL,
  NextToken = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **Filter**
  Named list where the name is the filter name and the value is the filter’s value.
  The value can be a vector or a list object (see below)[optional]

- **ReservedInstancesModificationId**
  List. IDs for the submitted modification request.[optional]

- **NextToken**
  Characters. The token for the next page of results[optional]
**simplify** Logical. Whether to simplify the result and handle `nextToken` in the response[optional]

**others** Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

**print_on_error** Logical. Whether to show an error message when a network error occurs.

**retry_time** Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

**network_timeout** Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

**region** Character. The region of the AWS service.

**Value**

A list object or a character vector

**Filter**

One or more filters.

- **client-token** - The idempotency token for the modification request.
- **create-date** - The time when the modification request was created.
- **effective-date** - The time when the modification becomes effective.
- **modification-result.reserved-instances-id** - The ID for the Reserved Instances created as part of the modification request. This ID is only available when the status of the modification is fulfilled.
- **modification-result.target-configuration.availability-zone** - The Availability Zone for the new Reserved Instances.
- **modification-result.target-configuration.instance-count** - The number of new Reserved Instances.
- **modification-result.target-configuration.instance-type** - The instance type of the new Reserved Instances.
- **modification-result.target-configuration.platform** - The network platform of the new Reserved Instances (EC2-Classic \ EC2-VPC).
- **reserved-instances-id** - The ID of the Reserved Instances modified.
- **reserved-instances-modification-id** - The ID of the modification request.
- **status** - The status of the Reserved Instances modification request (processing \ fulfilled \ failed).
- **status-message** - The reason for the status.
- **update-date** - The time when the modification request was last updated.

**ReservedInstancesModificationId**

IDs for the submitted modification request.
Describe Reserved Instances Offerings

Usage

```r
ec2_describe_reserved_instances_offerings(
  AvailabilityZone = NULL,
  Filter = NULL,
  IncludeMarketplace = NULL,
  InstanceType = NULL,
  MaxDuration = NULL,
  MaxInstanceCount = NULL,
  MinDuration = NULL,
  OfferingClass = NULL,
  ProductDescription = NULL,
  ReservedInstancesOfferingId = NULL,
  DryRun = NULL,
  InstanceTenancy = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  OfferingType = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

**AvailabilityZone**
Character. The Availability Zone in which the Reserved Instance can be used.[optional]

**Filter**
Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]

**IncludeMarketplace**
Logical. Include Reserved Instance Marketplace offerings in the response.[optional]

**InstanceType**
Character. The instance type that the reservation will cover (for example, `m1.small`). For more information,...[optional]

**MaxDuration**
Integer. The maximum duration (in seconds) to filter when searching for offerings.[optional]
MaxInstanceCount
Integer. The maximum number of instances to filter when searching for offerings. Default: 20 [optional]

MinDuration
Integer. The minimum duration (in seconds) to filter when searching for offerings.[optional]

OfferingClass
Character. The offering class of the Reserved Instance. Can be standard or convertible.[optional]

ProductDescription
Character. The Reserved Instance product platform description.[optional]

ReservedInstancesOfferingId
List. One or more Reserved Instances offering IDs.[optional]

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

InstanceTenancy
Character. The tenancy of the instances covered by the reservation.[optional]

MaxResults
Integer. The maximum number of results to return for the request in a single page.[optional]

NextToken
Characters. The token for the next page of results[optional]

OfferingType
Character. The Reserved Instance offering type.[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value
A list object or a character vector

AvailabilityZone
The Availability Zone in which the Reserved Instance can be used.

Filter
One or more filters.

• availability-zone - The Availability Zone where the Reserved Instance can be used.
• duration - The duration of the Reserved Instance (for example, one year or three years), in seconds (31536000 \ 94608000).
• **fixed-price** - The purchase price of the Reserved Instance (for example, 9800.0).
• **instance-type** - The instance type that is covered by the reservation.
• **marketplace** - Set to **true** to show only Reserved Instance Marketplace offerings. When this filter is not used, which is the default behavior, all offerings from both AWS and the Reserved Instance Marketplace are listed.
• **product-description** - The Reserved Instance product platform description. Instances that include (Amazon VPC) in the product platform description will only be displayed to EC2-Classic account holders and are for use with Amazon VPC. (Linux/UNIX|Linux/UNIX (Amazon VPC) \ SUSE Linux|SUSE Linux (Amazon VPC) \ Red Hat Enterprise Linux|Red Hat Enterprise Linux (Amazon VPC) \ Windows|Windows (Amazon VPC) \ Windows with SQL Server Standard|Windows with SQL Server Standard | Windows with SQL Server Web|Windows with SQL Server Web (Amazon VPC) \ Windows with SQL Server Enterprise|Windows with SQL Server Enterprise (Amazon VPC))
• **reserved-instances-offering-id** - The Reserved Instances offering ID.
• **scope** - The scope of the Reserved Instance (Availability Zone or Region).
• **usage-price** - The usage price of the Reserved Instance, per hour (for example, 0.84).

**IncludeMarketplace**

Include Reserved Instance Marketplace offerings in the response.

**InstanceType**

The instance type that the reservation will cover (for example, *m1.small*). For more information, see **Instance types** in the *Amazon EC2 User Guide*.

**MaxDuration**

The maximum duration (in seconds) to filter when searching for offerings.

Default: 94608000 (3 years)

**MaxInstanceCount**

The maximum number of instances to filter when searching for offerings.

Default: 20

**MinDuration**

The minimum duration (in seconds) to filter when searching for offerings.

Default: 2592000 (1 month)

**OfferingClass**

The offering class of the Reserved Instance. Can be **standard** or **convertible**.

**ProductDescription**

The Reserved Instance product platform description. Instances that include (Amazon VPC) in the description are for use with Amazon VPC.
ReservedInstancesOfferingId

One or more Reserved Instances offering IDs.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

InstanceTenancy

The tenancy of the instances covered by the reservation. A Reserved Instance with a tenancy of dedicated is applied to instances that run in a VPC on single-tenant hardware (i.e., Dedicated Instances).

Important: The host value cannot be used with this parameter. Use the default or dedicated values only.

Default: default

MaxResults

The maximum number of results to return for the request in a single page. The remaining results of the initial request can be seen by sending another request with the returned NextToken value. The maximum is 100.

Default: 100

OfferingType

The Reserved Instance offering type. If you are using tools that predate the 2011-11-01 API version, you only have access to the Medium Utilization Reserved Instance offering type.

Usage

c2_describe_route_tables(
    Filter = NULL,
    DryRun = NULL,
    RouteTableId = NULL,
    NextToken = NULL,
    MaxResults = NULL,
)
simplify = TRUE,
others = list(),
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

- **Filter**: Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below) [optional]
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request... [optional]
- **RouteTableId**: List. One or more route table IDs. Default: Describes all your route tables. [optional]
- **NextToken**: Characters. The token for the next page of results [optional]
- **MaxResults**: Integer. The maximum number of results to return with a single call. [optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response [optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

Filter

One or more filters.

- `association.route-table-association-id` - The ID of an association ID for the route table.
- `association.route-table-id` - The ID of the route table involved in the association.
- `association.subnet-id` - The ID of the subnet involved in the association.
- `association.main` - Indicates whether the route table is the main route table for the VPC (true \ false). Route tables that do not have an association ID are not returned in the response.
• **owner-id** - The ID of the AWS account that owns the route table.
• **route-table-id** - The ID of the route table.
• **route.destination-cidr-block** - The IPv4 CIDR range specified in a route in the table.
• **route.destination-ipv6-cidr-block** - The IPv6 CIDR range specified in a route in the route table.
• **route.destination-prefix-list-id** - The ID (prefix) of the AWS service specified in a route in the table.
• **route.egress-only-internet-gateway-id** - The ID of an egress-only Internet gateway specified in a route in the route table.
• **route.gateway-id** - The ID of a gateway specified in a route in the table.
• **route.instance-id** - The ID of an instance specified in a route in the table.
• **route.nat-gateway-id** - The ID of a NAT gateway.
• **route.transit-gateway-id** - The ID of a transit gateway.
• **route.origin** - Describes how the route was created. **CreateRouteTable** indicates that the route was automatically created when the route table was created; **CreateRoute** indicates that the route was manually added to the route table; **EnableVgwRoutePropagation** indicates that the route was propagated by route propagation.
• **route.state** - The state of a route in the route table (active | blackhole). The blackhole state indicates that the route's target isn't available (for example, the specified gateway isn't attached to the VPC, the specified NAT instance has been terminated, and so on).
• **route.vpc-peering-connection-id** - The ID of a VPC peering connection specified in a route in the table.
• **tag:** - The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key **Owner** and the value **TeamA**, specify tag:Owner for the filter name and TeamA for the filter value.
• **tag-key** - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.
• **vpc-id** - The ID of the VPC for the route table.

**DryRun**
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is **DryRunOperation**. Otherwise, it is **UnauthorizedOperation**.

**RouteTableId**
One or more route table IDs.
Default: Describes all your route tables.

**MaxResults**
The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned **nextToken** value.
ec2\_describe\_scheduled\_instances

*Describe Scheduled Instances*

**Description**

Describes the specified Scheduled Instances or all your Scheduled Instances.

**Usage**

```r
ec2\_describe\_scheduled\_instances(
    DryRun = \text{NULL},
    Filter = \text{NULL},
    MaxResults = \text{NULL},
    NextToken = \text{NULL},
    ScheduledInstanceId = \text{NULL},
    SlotStartTimeRange = \text{NULL},
    simplify = \text{TRUE},
    others = \text{list()},
    print\_on\_error = aws\_get\_print\_on\_error(),
    retry\_time = aws\_get\_retry\_time(),
    network\_timeout = aws\_get\_network\_timeout(),
    region = aws\_get\_region()
)
```

**Arguments**

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]

- **Filter**
  Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below). [optional]

- **MaxResults**
  Integer. The maximum number of results to return in a single call. [optional]

- **NextToken**
  Characters. The token for the next page of results. [optional]

- **ScheduledInstanceId**
  List. The Scheduled Instance IDs. [optional]

- **SlotStartTimeRange**
  Object. The time period for the first schedule to start. [optional]

- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]

- **print\_on\_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry\_time**
  Integer. Number of retries for a REST request when encountering the network issue. If the request has been sent `retry\_time` times but still not able to get the response, an error will be thrown.
network_timeout
    Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
    Character. The region of the AWS service.

Value
    A list object or a character vector

DryRun
    Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Filter
    The filters.
    - availability-zone - The Availability Zone (for example, us-west-2a).
    - instance-type - The instance type (for example, c4.large).
    - network-platform - The network platform (EC2-Classic or EC2-VPC).
    - platform - The platform (Linux/UNIX or Windows).

MaxResults
    The maximum number of results to return in a single call. This value can be between 5 and 300. The default value is 100. To retrieve the remaining results, make another call with the returned NextToken value.

ScheduledInstanceId
    The Scheduled Instance IDs.

SlotStartTimeRange
    The time period for the first schedule to start.
ec2_describe_scheduled_instance_availability

Describe Scheduled Instance Availability

Description

Describe Scheduled Instance Availability

Usage

```
ec2_describe_scheduled_instance_availability(
    FirstSlotStartTimeRange,
    Recurrence,
    DryRun = NULL,
    Filter = NULL,
    MaxResults = NULL,
    MaxSlotDurationInHours = NULL,
    MinSlotDurationInHours = NULL,
    NextToken = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **FirstSlotStartTimeRange**
  Object. The time period for the first schedule to start.

- **Recurrence**
  Object. The schedule recurrence.

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

- **Filter**
  Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]

- **MaxResults**
  Integer. The maximum number of results to return in a single call.[optional]

- **MaxSlotDurationInHours**
  Integer. The maximum available duration, in hours.[optional]

- **MinSlotDurationInHours**
  Integer. The minimum available duration, in hours.[optional]

- **NextToken**
  Characters. The token for the next page of results[optional]

- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.

Value
A list object or a character vector

FirstSlotStartTimeRange
The time period for the first schedule to start.

Recurrence
The schedule recurrence.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Filter
The filters.

• availability-zone - The Availability Zone (for example, us-west-2a).
• instance-type - The instance type (for example, c4.large).
• network-platform - The network platform (EC2-Classic or EC2-VPC).
• platform - The platform (Linux/UNIX or Windows).

MaxResults
The maximum number of results to return in a single call. This value can be between 5 and 300. The default value is 300. To retrieve the remaining results, make another call with the returned NextToken value.

MaxSlotDurationInHours
The maximum available duration, in hours. This value must be greater than MinSlotDurationInHours and less than 1,720.
MinSlotDurationInHours

The minimum available duration, in hours. The minimum required duration is 1,200 hours per year. For example, the minimum daily schedule is 4 hours, the minimum weekly schedule is 24 hours, and the minimum monthly schedule is 100 hours.

ec2_describe_security_groups

Describe Security Groups

Description

Describe Security Groups

Usage

ect2_describe_security_groups(
  Filter = NULL,
  GroupId = NULL,
  GroupName = NULL,
  DryRun = NULL,
  NextToken = NULL,
  MaxResults = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

Filter Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]

GroupId List. The IDs of the security groups.[optional]

GroupName List. [EC2-Classic and default VPC only] The names of the security groups.[optional]

DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]

NextToken Characters. The token for the next page of results[optional]

MaxResults Integer. The maximum number of results to return in a single call.[optional]

simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]

others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  Character. The region of the AWS service.

Value
A list object or a character vector

Filter
The filters. If using multiple filters for rules, the results include security groups for which any combination of rules - not necessarily a single rule - match all filters.

• description - The description of the security group.
• egress.ip-permission.cidr - An IPv4 CIDR block for an outbound security group rule.
• egress.ip-permission.from-port - For an outbound rule, the start of port range for the TCP and UDP protocols, or an ICMP type number.
• egress.ip-permission.group-id - The ID of a security group that has been referenced in an outbound security group rule.
• egress.ip-permission.group-name - The name of a security group that is referenced in an outbound security group rule.
• egress.ip-permission.ipv6-cidr - An IPv6 CIDR block for an outbound security group rule.
• egress.ip-permission.prefix-list-id - The ID of a prefix list to which a security group rule allows outbound access.
• egress.ip-permission.protocol - The IP protocol for an outbound security group rule (tcp | udp | icmp, a protocol number, or -1 for all protocols).
• egress.ip-permission.to-port - For an outbound rule, the end of port range for the TCP and UDP protocols, or an ICMP code.
• egress.ip-permission.user-id - The ID of an AWS account that has been referenced in an outbound security group rule.
• group-id - The ID of the security group.
• group-name - The name of the security group.
• ip-permission.cidr - An IPv4 CIDR block for an inbound security group rule.
• ip-permission.from-port - For an inbound rule, the start of port range for the TCP and UDP protocols, or an ICMP type number.
• ip-permission.group-id - The ID of a security group that has been referenced in an inbound security group rule.
• ip-permission.group-name - The name of a security group that is referenced in an inbound security group rule.
• ip-permission.ipv6-cidr - An IPv6 CIDR block for an inbound security group rule.
• ip-permission.prefix-list-id - The ID of a prefix list from which a security group rule allows inbound access.
• ip-permission.protocol - The IP protocol for an inbound security group rule (tcp \| udp \| icmp, a protocol number, or -1 for all protocols).
• ip-permission.to-port - For an inbound rule, the end of port range for the TCP and UDP protocols, or an ICMP code.
• ip-permission.user-id - The ID of an AWS account that has been referenced in an inbound security group rule.
• owner-id - The AWS account ID of the owner of the security group.
• tag: - The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA for the filter value.
• tag-key - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.
• vpc-id - The ID of the VPC specified when the security group was created.

** groupId **

The IDs of the security groups. Required for security groups in a nondefault VPC.
Default: Describes all your security groups.

** group-name **

[EC2-Classic and default VPC only] The names of the security groups. You can specify either the security group name or the security group ID. For security groups in a nondefault VPC, use the group-name filter to describe security groups by name.
Default: Describes all your security groups.

** dry-run **

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

** max-results **

The maximum number of results to return in a single call. To retrieve the remaining results, make another request with the returned NextToken value. This value can be between 5 and 1000. If this parameter is not specified, then all results are returned.
ec2_describe_security_group_references

Describe Security Group References

Description

[VPC only] Describes the VPCs on the other side of a VPC peering connection that are referencing the security groups you’ve specified in this request.

Usage

ec2_describe_security_group_references(
    GroupId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

GroupId List. The IDs of the security groups in your account.
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value

A list object or a character vector

GroupId

The IDs of the security groups in your account.
ec2_describe_snapshots

Describe Snapshots

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_describe_snapshots

Describe Snapshots

Usage

ey2_describe_snapshots(
    Filter = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    Owner = NULL,
    RestorableBy = NULL,
    SnapshotId = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

Filter Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]
MaxResults Integer. The maximum number of snapshot results returned by DescribeSnapshots in paginated output.[optional]
NextToken Characters. The token for the next page of results.[optional]
Owner List. Scopes the results to snapshots with the specified owners.[optional]
RestorableBy List. The IDs of the AWS accounts that can create volumes from the snapshot.[optional]
SnapshotId List. The snapshot IDs.[optional]
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value
A list object or a character vector

Filter
The filters.

• description - A description of the snapshot.
• encrypted - Indicates whether the snapshot is encrypted (true \| false)
• owner-alias - The owner alias, from an Amazon-maintained list (amazon). This is not the user-configured AWS account alias set using the IAM console. We recommend that you use the related parameter instead of this filter.
• owner-id - The AWS account ID of the owner. We recommend that you use the related parameter instead of this filter.
• progress - The progress of the snapshot, as a percentage (for example, 80\%
• snapshot-id - The snapshot ID.
• start-time - The time stamp when the snapshot was initiated.
• status - The status of the snapshot (pending \| completed \| error).
• tag:\- The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA for the filter value.
• tag-key - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.
• volume-id - The ID of the volume the snapshot is for.
• volume-size - The size of the volume, in GiB.
MaxResults

The maximum number of snapshot results returned by DescribeSnapshots in paginated output. When this parameter is used, DescribeSnapshots only returns MaxResults results in a single page along with a NextToken response element. The remaining results of the initial request can be seen by sending another DescribeSnapshots request with the returned NextToken value. This value can be between 5 and 1,000; if MaxResults is given a value larger than 1,000, only 1,000 results are returned. If this parameter is not used, then DescribeSnapshots returns all results. You cannot specify this parameter and the snapshot IDs parameter in the same request.

Owner

Scopes the results to snapshots with the specified owners. You can specify a combination of AWS account IDs, self, and amazon.

RestorableBy

The IDs of the AWS accounts that can create volumes from the snapshot.

SnapshotId

The snapshot IDs.

Default: Describes the snapshots for which you have create volume permissions.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_describe_snapshot_attribute

Describe Snapshot Attribute

Description

Describe Snapshot Attribute

Usage

ec2_describe_snapshot_attribute(
    Attribute,
    SnapshotId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
)
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

Attribute Character. The snapshot attribute you would like to view.
SnapshotId Character. The ID of the EBS snapshot.
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value

A list object or a character vector

Attribute

The snapshot attribute you would like to view.

SnapshotId

The ID of the EBS snapshot.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
ec2_describe_spot_datafeed_subscription

Describe Spot Datafeed Subscription

Description

Describes the data feed for Spot Instances. For more information, see Spot Instance data feed in the Amazon EC2 User Guide for Linux Instances.

Usage

ec2_describe_spot_datafeed_subscription(
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request, ...[optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response[optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>

Value

A list object or a character vector

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
ec2\_describe\_spot\_fleet\_instances

Describe Spot Fleet Instances

Description

Describes the running instances for the specified Spot Fleet.

Usage

```r
ec2\_describe\_spot\_fleet\_instances(
    SpotFleetRequestId,
    DryRun = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    simplify = TRUE,
    others = list(),
    print\_on\_error = aws\_get\_print\_on\_error(),
    retry\_time = aws\_get\_retry\_time(),
    network\_timeout = aws\_get\_network\_timeout(),
    region = aws\_get\_region()
)
```

Arguments

- **SpotFleetRequestId**: Character. The ID of the Spot Fleet request.
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request. Optional.
- **MaxResults**: Integer. The maximum number of results to return in a single call. Optional.
- **NextToken**: Characters. The token for the next page of results. Optional.
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response. Optional.
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request. Optional.
- **print\_on\_error**: Logical. Whether to show an error message when a network error occurs.
- **retry\_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry\_time` times but still not be able to get the response, an error will be thrown.
- **network\_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.
ec2_describe_spot_fleet_requests

Value

A list object or a character vector

SpotFleetRequestId

The ID of the Spot Fleet request.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

MaxResults

The maximum number of results to return in a single call. Specify a value between 1 and 1000. The default value is 1000. To retrieve the remaining results, make another call with the returned NextToken value.

ec2_describe_spot_fleet_requests

Describe Spot Fleet Requests

Description

Describe Spot Fleet Requests

Usage

ec2_describe_spot_fleet_requests(
  DryRun = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  SpotFleetRequestId = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region())
ec2_describe_spot_fleet_requests

Arguments

DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request, [optional]

MaxResults Integer. The maximum number of results to return in a single call [optional]

NextToken Characters. The token for the next page of results [optional]

SpotFleetRequestId List. The IDs of the Spot Fleet requests [optional]

simplify Logical. Whether to simplify the result and handle nextToken in the response [optional]

others Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.

Value

A list object or a character vector

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

MaxResults

The maximum number of results to return in a single call. Specify a value between 1 and 1000. The default value is 1000. To retrieve the remaining results, make another call with the returned NextToken value.

SpotFleetRequestId

The IDs of the Spot Fleet requests.
ec2_describe_spot_fleet_request_history

Describe Spot Fleet Request History

Description

Describe Spot Fleet Request History

Usage

ec2_describe_spot_fleet_request_history(
    SpotFleetRequestId,
    StartTime,
    DryRun = NULL,
    EventType = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

SpotFleetRequestId
    Character. The ID of the Spot Fleet request.

StartTime
    Character. The starting date and time for the events, in UTC format (for example, YYYY-MM-DDTHH:MM:SSZ)....

DryRun
    Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]

EventType
    Character. The type of events to describe. By default, all events are described.[optional]

MaxResults
    Integer. The maximum number of results to return in a single call.[optional]

NextToken
    Characters. The token for the next page of results[optional]

simplify
    Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
    Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
    Logical. Whether to show an error message when a network error occurs.

retry_time
    Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout

    Numeric. Number of seconds to wait for a REST response until giving up. Can
    not be less than 1 ms.

region

    Character. The region of the AWS service.

Value

    A list object or a character vector

SpotFleetRequestId

    The ID of the Spot Fleet request.

StartTime

    The starting date and time for the events, in UTC format (for example, YYYY-MM-DDTHH:MM:SSZ).

DryRun

    Checks whether you have the required permissions for the action, without actually making the
    request, and provides an error response. If you have the required permissions, the error response is
    DryRunOperation. Otherwise, it is UnauthorizedOperation.

EventType

    The type of events to describe. By default, all events are described.

MaxResults

    The maximum number of results to return in a single call. Specify a value between 1 and 1000.
    The default value is 1000. To retrieve the remaining results, make another call with the returned
    NextToken value.
Usage

```r
ec2_describe_spot_instance_requests(
    Filter = NULL,
    DryRun = NULL,
    SpotInstanceRequestId = NULL,
    NextToken = NULL,
    MaxResults = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **Filter**: Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below) [optional]
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request... [optional]
- **SpotInstanceRequestId**: List. One or more Spot Instance request IDs. [optional]
- **NextToken**: Characters. The token for the next page of results [optional]
- **MaxResults**: Integer. The maximum number of results to return in a single call. [optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response [optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

Filter

One or more filters.

- availability-zone-group - The Availability Zone group.
• create-time - The time stamp when the Spot Instance request was created.
• fault-code - The fault code related to the request.
• fault-message - The fault message related to the request.
• instance-id - The ID of the instance that fulfilled the request.
• launch-group - The Spot Instance launch group.
• launch.block-device-mapping.delete-on-termination - Indicates whether the EBS volume is deleted on instance termination.
• launch.block-device-mapping.device-name - The device name for the volume in the block device mapping (for example, /dev/sdh or xvdh).
• launch.block-device-mapping.snapshot-id - The ID of the snapshot for the EBS volume.
• launch.block-device-mapping.volume-size - The size of the EBS volume, in GiB.
• launch.block-device-mapping.volume-type - The type of EBS volume: gp2 for General Purpose SSD, io1 or io2 for Provisioned IOPS SSD, st1 for Throughput Optimized HDD, sc1 for Cold HDD, or standard for Magnetic.
• launch.group-id - The ID of the security group for the instance.
• launch.group-name - The name of the security group for the instance.
• launch.image-id - The ID of the AMI.
• launch.instance-type - The type of instance (for example, m3.medium).
• launch.kernel-id - The kernel ID.
• launch.key-name - The name of the key pair the instance launched with.
• launch.monitoring-enabled - Whether detailed monitoring is enabled for the Spot Instance.
• launch.ramdisk-id - The RAM disk ID.
• launched-availability-zone - The Availability Zone in which the request is launched.
• network-interface.addresses.primary - Indicates whether the IP address is the primary private IP address.
• network-interface.delete-on-termination - Indicates whether the network interface is deleted when the instance is terminated.
• network-interface.description - A description of the network interface.
• network-interface.device-index - The index of the device for the network interface attachment on the instance.
• network-interface.group-id - The ID of the security group associated with the network interface.
• network-interface.network-interface-id - The ID of the network interface.
• network-interface.private-ip-address - The primary private IP address of the network interface.
• network-interface.subnet-id - The ID of the subnet for the instance.
• product-description - The product description associated with the instance (Linux/UNIX \ Windows).
**ec2_describe_spot_price_history**

*spot-instance-request-id* - The Spot Instance request ID.

*spot-price* - The maximum hourly price for any Spot Instance launched to fulfill the request.

*state* - The state of the Spot Instance request (open \| active \| closed \| cancelled \| failed). Spot request status information can help you track your Amazon EC2 Spot Instance requests. For more information, see Spot request status in the *Amazon EC2 User Guide for Linux Instances*.

*status-code* - The short code describing the most recent evaluation of your Spot Instance request.

*status-message* - The message explaining the status of the Spot Instance request.

*tag:* - The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key *Owner* and the value *TeamA*, specify *tag:Owner* for the filter name and *TeamA* for the filter value.

*tag-key* - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.

*type* - The type of Spot Instance request (one-time \| persistent).

*valid-from* - The start date of the request.

*valid-until* - The end date of the request.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is *DryRunOperation*. Otherwise, it is *UnauthorizedOperation*.

**SpotInstanceRequestIds**

One or more Spot Instance request IDs.

**MaxResults**

The maximum number of results to return in a single call. Specify a value between 5 and 1000. To retrieve the remaining results, make another call with the returned *NextToken* value.
ec2_describe_spot_price_history

Usage

ec2_describe_spot_price_history(
    Filter = NULL,
    AvailabilityZone = NULL,
    DryRun = NULL,
    EndTime = NULL,
    InstanceType = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    ProductDescription = NULL,
    StartTime = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

Filter  Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]
AvailabilityZone  Character. Filters the results by the specified Availability Zone.[optional]
DryRun  Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]
EndTime  Character. The date and time, up to the current date, from which to stop retrieving the price history data....[optional]
InstanceType  List. Filters the results by the specified instance types.[optional]
MaxResults  Integer. The maximum number of results to return in a single call.[optional]
NextToken  Characters. The token for the next page of results[optional]
ProductDescription  List. Filters the results by the specified basic product descriptions.[optional]
StartTime  Character. The date and time, up to the past 90 days, from which to start retrieving the price history data....[optional]
simplify  Logical. Whether to simplify the result and handle nextToken in the response[optional]
others  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error  Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout
   Numeric. Number of seconds to wait for a REST response until giving up. Can
   not be less than 1 ms.

region
   Character. The region of the AWS service.

Value
   A list object or a character vector

Filter
   One or more filters.
   
   • availability-zone - The Availability Zone for which prices should be returned.
   • instance-type - The type of instance (for example, m3.medium).
   • product-description - The product description for the Spot price (Linux/UNIX \ Red Hat Enterprise Linux \ SUSE Linux \ Windows \ Linux/UNIX (Amazon VPC) \ Red Hat Enterprise Linux (Amazon VPC) \ SUSE Linux (Amazon VPC) \ Windows (Amazon VPC)).
   • spot-price - The Spot price. The value must match exactly (or use wildcards; greater than
     or less than comparison is not supported).
   • timestamp - The time stamp of the Spot price history, in UTC format (for example, YYYY-MM-DDTHH:MM:SSZ). You can use wildcards (\* and ?). Greater than or less than comparison is not supported.

AvailabilityZone
   Filters the results by the specified Availability Zone.

DryRun
   Checks whether you have the required permissions for the action, without actually making the
   request, and provides an error response. If you have the required permissions, the error response is
   DryRunOperation. Otherwise, it is UnauthorizedOperation.

EndTime
   The date and time, up to the current date, from which to stop retrieving the price history data, in
   UTC format (for example, YYYY-MM-DDTHH:MM:SSZ).

InstanceType
   Filters the results by the specified instance types.

MaxResults
   The maximum number of results to return in a single call. Specify a value between 1 and 1000.
   The default value is 1000. To retrieve the remaining results, make another call with the returned
   NextToken value.
**ProductDescription**

Filters the results by the specified basic product descriptions.

**StartTime**

The date and time, up to the past 90 days, from which to start retrieving the price history data, in UTC format (for example, `YYYY-MM-DDTHH:MM:SSZ`).

---

**ec2_describe_stale_security_groups**

*Describe Stale Security Groups*

---

**Description**

[VPC only] Describes the stale security group rules for security groups in a specified VPC. Rules are stale when they reference a deleted security group in a peer VPC, or a security group in a peer VPC for which the VPC peering connection has been deleted.

**Usage**

```r
ec2_describe_stale_security_groups(
  VpcId,
  DryRun = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- **VpcId**
  - Character. The ID of the VPC.
- **DryRun**
  - Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
- **MaxResults**
  - Integer. The maximum number of items to return for this request.[optional]
- **NextToken**
  - Characters. The token for the next page of results.[optional]
- **simplify**
  - Logical. Whether to simplify the result and handle `nextToken` in the response.[optional]
- **others**
  - Named list. The parameters that are not included in the function parameters and need to be added into the request.[optional]
- **print_on_error**
  - Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  Character. The region of the AWS service.

Value
A list object or a character vector

VpcId
The ID of the VPC.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

MaxResults
The maximum number of items to return for this request. The request returns a token that you can specify in a subsequent call to get the next set of results.

describe_subnets  Describe Subnets

Description
Describe Subnets

Usage
```r
describe_subnets(  
  Filter = NULL,  
  SubnetId = NULL,  
  DryRun = NULL,  
  NextToken = NULL,  
  MaxResults = NULL,  
  simplify = TRUE,  
  others = list(),  
  print_on_error = aws_get_print_on_error(),  
  retry_time = aws_get_retry_time(),  
  network_timeout = aws_get_network_timeout(),  
  region = aws_get_region()  
)
```
ec2_describe_subnets

Arguments

Filter  Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]
SubnetId List. One or more subnet IDs. Default: Describes all your subnets. [optional]
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]
NextToken Characters. The token for the next page of results[optional]
MaxResults Integer. The maximum number of results to return with a single call.[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value

A list object or a character vector

Filter

One or more filters.

• availability-zone - The Availability Zone for the subnet. You can also use availabilityZone as the filter name.
• availability-zone-id - The ID of the Availability Zone for the subnet. You can also use availabilityZoneId as the filter name.
• available-ip-address-count - The number of IPv4 addresses in the subnet that are available.
• cidr-block - The IPv4 CIDR block of the subnet. The CIDR block you specify must exactly match the subnet's CIDR block for information to be returned for the subnet. You can also use cidr or cidrBlock as the filter names.
• default-for-az - Indicates whether this is the default subnet for the Availability Zone. You can also use defaultForAz as the filter name.
• ipv6-cidr-block-association.ipv6-cidr-block - An IPv6 CIDR block associated with the subnet.
• ipv6-cidr-block-association.association-id - An association ID for an IPv6 CIDR block associated with the subnet.
• ipv6-cidr-block-association.state - The state of an IPv6 CIDR block associated with the subnet.
• owner-id - The ID of the AWS account that owns the subnet.
• state - The state of the subnet (pending \| available).
• subnet-arn - The Amazon Resource Name (ARN) of the subnet.
• subnet-id - The ID of the subnet.
• tag\: - The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA for the filter value.
• tag\=key - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.
• vpc-id - The ID of the VPC for the subnet.

SubnetId

One or more subnet IDs.
Default: Describes all your subnets.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

MaxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

---

**ec2_describe_tags**  
Describe Tags

**Description**

Describe Tags

**Usage**

```r
ec2_describe_tags(
  DryRun = NULL,
  Filter = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  simplify = TRUE,
)```
others = list(),
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
Filter Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]
MaxResults Integer. The maximum number of results to return in a single call.[optional]
NextToken Characters. The token for the next page of results[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value
A list object or a character vector

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Filter
The filters.
  • key - The tag key.
  • resource-id - The ID of the resource.
  • resource-type - The resource type (customer-gateway\ dedicated-host\ dhcp-options\ elastic-ip\ fleet\ fpga-image\ host-reservation\ image\ instance\ internet-gateway\ key-pair\ launch-template\ natgateway\ network-acl\ network-interface\
• tag: The key/value combination of the tag. For example, specify `tag:Owner` for the filter name and `TeamA` for the filter value to find resources with the tag `Owner=TeamA`.

MaxResults

The maximum number of results to return in a single call. This value can be between 5 and 1000. To retrieve the remaining results, make another call with the returned NextToken value.

describe_traffic_mirror_filters

Describe Traffic Mirror Filters

Description

Describes one or more Traffic Mirror filters.

Usage

```r
describe_traffic_mirror_filters(  TrafficMirrorFilterId = NULL,  DryRun = NULL,  Filter = NULL,  MaxResults = NULL,  NextToken = NULL,  simplify = TRUE,  others = list(),  print_on_error = aws_get_print_on_error(),  retry_time = aws_get_retry_time(),  network_timeout = aws_get_network_timeout(),  region = aws_get_region()
)
```

Arguments

- **TrafficMirrorFilterId**: List. The ID of the Traffic Mirror filter.[optional]
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
- **Filter**: Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below).[optional]
- **MaxResults**: Integer. The maximum number of results to return with a single call.[optional]
- **NextToken**: Characters. The token for the next page of results[optional]
`ec2_describe_traffic_mirror_filters`

- **simplify** Logical. Whether to simplify the result and handle `nextToken` in the response [optional]

- **others** Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]

- **print_on_error** Logical. Whether to show an error message when a network error occurs.

- **retry_time** Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

- **network_timeout** Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region** Character. The region of the AWS service.

### Value

A list object or a character vector

### TrafficMirrorFilterId

The ID of the Traffic Mirror filter.

### DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

### Filter

One or more filters. The possible values are:

- **description** The Traffic Mirror filter description.
- **traffic-mirror-filter-id** The ID of the Traffic Mirror filter.

### MaxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned `nextToken` value.
ec2_describe_traffic_mirror_sessions

Describe Traffic Mirror Sessions

Description

Describes one or more Traffic Mirror sessions. By default, all Traffic Mirror sessions are described. Alternatively, you can filter the results.

Usage

ec2_describe_traffic_mirror_sessions(
    TrafficMirrorSessionId = NULL,
    DryRun = NULL,
    Filter = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

TrafficMirrorSessionId
List. The ID of the Traffic Mirror session.[optional]

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

Filter
Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below).[optional]

MaxResults
Integer. The maximum number of results to return with a single call.[optional]

NextToken
Characters. The token for the next page of results[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.
Value

A list object or a character vector

TrafficMirrorSessionId

The ID of the Traffic Mirror session.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Filter

One or more filters. The possible values are:

- description: The Traffic Mirror session description.
- network-interface-id: The ID of the Traffic Mirror session network interface.
- owner-id: The ID of the account that owns the Traffic Mirror session.
- packet-length: The assigned number of packets to mirror.
- session-number: The assigned session number.
- traffic-mirror-filter-id: The ID of the Traffic Mirror filter.
- traffic-mirror-session-id: The ID of the Traffic Mirror session.
- traffic-mirror-target-id: The ID of the Traffic Mirror target.
- virtual-network-id: The virtual network ID of the Traffic Mirror session.

MaxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

Description

Information about one or more Traffic Mirror targets.
Usage

```r
ec2_describe_traffic_mirror_targets(
  TrafficMirrorTargetId = NULL,
  DryRun = NULL,
  Filter = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  simplify = TRUE,
  others = list()
)
```

Arguments

- **TrafficMirrorTargetId**
  List. The ID of the Traffic Mirror targets.[optional]

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]

- **Filter**
  Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]

- **MaxResults**
  Integer. The maximum number of results to return with a single call.[optional]

- **NextToken**
  Characters. The token for the next page of results[optional]

- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  Character. The region of the AWS service.

Value

A list object or a character vector

**TrafficMirrorTargetId**

The ID of the Traffic Mirror targets.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Filter

One or more filters. The possible values are:

- description: The Traffic Mirror target description.
- network-interface-id: The ID of the Traffic Mirror session network interface.
- network-load-balancer-arn: The Amazon Resource Name (ARN) of the Network Load Balancer that is associated with the session.
- owner-id: The ID of the account that owns the Traffic Mirror session.
- traffic-mirror-target-id: The ID of the Traffic Mirror target.

MaxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

---

ec2_describe_transit_gateways

*Describe Transit Gateways*

---

Description

Describes one or more transit gateways. By default, all transit gateways are described. Alternatively, you can filter the results.

Usage

```r
ec2_describe_transit_gateways(
    TransitGatewayIds = NULL,
    Filter = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```
Arguments

- **TransitGatewayIds**
  List. The IDs of the transit gateways. [optional]

- **Filter**
  Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below). [optional]

- **MaxResults**
  Integer. The maximum number of results to return with a single call. [optional]

- **NextToken**
  Characters. The token for the next page of results. [optional]

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]

- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  Character. The region of the AWS service.

Value

A list object or a character vector

**TransitGatewayIds**

The IDs of the transit gateways.

**Filter**

One or more filters. The possible values are:

- **options.propagation-default-route-table-id** - The ID of the default propagation route table.
- **options.amazon-side-asn** - The private ASN for the Amazon side of a BGP session.
- **options.association-default-route-table-id** - The ID of the default association route table.
- **options.auto-accept-shared-attachments** - Indicates whether there is automatic acceptance of attachment requests (enable | disable).
- **options.default-route-table-association** - Indicates whether resource attachments are automatically associated with the default association route table (enable | disable).
- **options.default-route-table-propagation** - Indicates whether resource attachments automatically propagate routes to the default propagation route table (enable | disable).
• **options.dns-support** - Indicates whether DNS support is enabled (enable \| disable).
• **options.vpn-ecmp-support** - Indicates whether Equal Cost Multipath Protocol support is enabled (enable \| disable).
• **owner-id** - The ID of the AWS account that owns the transit gateway.
• **state** - The state of the transit gateway (available \| deleted \| deleting \| modifying \| pending).
• **transit-gateway-id** - The ID of the transit gateway.

**MaxResults**

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned `nextToken` value.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

```
ec2_describe_transit_gateway_attachments

Describe Transit Gateway Attachments

Description

Describes one or more attachments between resources and transit gateways. By default, all attachments are described. Alternatively, you can filter the results by attachment ID, attachment state, resource ID, or resource owner.

Usage

ec2_describe_transit_gateway_attachments(
    TransitGatewayAttachmentIds = NULL,
    Filter = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```
Arguments

TransitGatewayAttachmentIds
List. The IDs of the attachments.[optional]

Filter
Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below).[optional]

MaxResults
Integer. The maximum number of results to return with a single call.[optional]

NextToken
Characters. The token for the next page of results[optional]

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value

A list object or a character vector

TransitGatewayAttachmentIds

The IDs of the attachments.

Filter

One or more filters. The possible values are:

- association.state - The state of the association (associating \ associating \ disassociating).
- association.transit-gateway-route-table-id - The ID of the route table for the transit gateway.
- resource-id - The ID of the resource.
- resource-owner-id - The ID of the AWS account that owns the resource.
- resource-type - The resource type. Valid values are vpc \ vpn \ direct-connect-gateway \ peering \ connect.
- state - The state of the attachment. Valid values are available \ deleted \ deleting \ failed \ failing \ initiatingRequest \ modifying \ pendingAcceptance \ pending \ rollingBack \ rejected \ rejecting.
- transit-gateway-attachment-id - The ID of the attachment.
- transit-gateway-id - The ID of the transit gateway.
- transit-gateway-owner-id - The ID of the AWS account that owns the transit gateway.
MaxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**ec2_describe_transit_gateway_connects**

*Describe Transit Gateway Connects*

---

**Description**

Describes one or more Connect attachments.

**Usage**

```r
ec2_describe_transit_gateway_connects(
  TransitGatewayAttachmentIds = NULL,
  Filter = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- **TransitGatewayAttachmentIds**
  - List. The IDs of the attachments.[optional]
- **Filter**
  - Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below).[optional]
- **MaxResults**
  - Integer. The maximum number of results to return with a single call.[optional]
- **NextToken**
  - Characters. The token for the next page of results[optional]
- **DryRun**
  - Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]
- **simplify**
  - Logical. Whether to simplify the result and handle nextToken in the response[optional]
ec2_describe_transit_gateway_connects

others
   Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]

print_on_error
   Logical. Whether to show an error message when a network error occurs.

retry_time
   Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
   Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
   Character. The region of the AWS service.

Value
   A list object or a character vector

TransitGatewayAttachmentIds
   The IDs of the attachments.

Filter
   One or more filters. The possible values are:
   • options.protocol - The tunnel protocol (gre).
   • state - The state of the attachment (initiating | initiatingRequest | pendingAcceptance | rollingBack | pending | available | modifying | deleting | deleted | failed | rejected | rejecting | failing).
   • transit-gateway-attachment-id - The ID of the Connect attachment.
   • transit-gateway-id - The ID of the transit gateway.
   • transport-transit-gateway-attachment-id - The ID of the transit gateway attachment from which the Connect attachment was created.

MaxResults
   The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

DryRun
   Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
Describe Transit Gateway Connect Peers

Describes one or more Connect peers.

Usage

```r
ec2_describe_transit_gateway_connect_peers(
  TransitGatewayConnectPeerIds = NULL,
  Filter = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **TransitGatewayConnectPeerIds**
  - List. The IDs of the Connect peers.[optional]
- **Filter**
  - Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]
- **MaxResults**
  - Integer. The maximum number of results to return with a single call.[optional]
- **NextToken**
  - Characters. The token for the next page of results[optional]
- **DryRun**
  - Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]
- **simplify**
  - Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others**
  - Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**
  - Logical. Whether to show an error message when a network error occurs.
- **retry_time**
  - Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- **network_timeout**
  - Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**
  - Character. The region of the AWS service.
Value

A list object or a character vector

TransitGatewayConnectPeerIds

The IDs of the Connect peers.

Filter

One or more filters. The possible values are:

• state - The state of the Connect peer (pending | available | deleting | deleted).
• transit-gateway-attachment-id - The ID of the attachment.
• transit-gateway-connect-peer-id - The ID of the Connect peer.

MaxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_describe_transit_gateway_multicast_domains

Describe Transit Gateway Multicast Domains

Description

Describes one or more transit gateway multicast domains.

Usage

```r
ec2_describe_transit_gateway_multicast_domains(
  TransitGatewayMulticastDomainIds = NULL,
  Filter = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
Arguments

TransitGatewayMulticastDomainIds  
List. The ID of the transit gateway multicast domain.[optional]

Filter  
Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]

MaxResults  
Integer. The maximum number of results to return with a single call.[optional]

NextToken  
Characters. The token for the next page of results[optional]

DryRun  
Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]

simplify  
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others  
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error  
Logical. Whether to show an error message when a network error occurs.

retry_time  
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  
Character. The region of the AWS service.

Value

A list object or a character vector

TransitGatewayMulticastDomainIds

The ID of the transit gateway multicast domain.

Filter

One or more filters. The possible values are:

- state - The state of the transit gateway multicast domain. Valid values are pending \ available \ deleting \ deleted.
- transit-gateway-id - The ID of the transit gateway.
- transit-gateway-multicast-domain-id - The ID of the transit gateway multicast domain.

MaxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**ec2_describe_transit_gateway_peering_attachments**

*Describe Transit Gateway Peering Attachments*

---

**Description**

Describes your transit gateway peering attachments.

**Usage**

```r
ec2_describe_transit_gateway_peering_attachments(
    TransitGatewayAttachmentIds = NULL,
    Filter = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

**Arguments**

- **TransitGatewayAttachmentIds**
  - List. One or more IDs of the transit gateway peering attachments.[optional]
- **Filter**
  - Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]
- **MaxResults**
  - Integer. The maximum number of results to return with a single call.[optional]
- **NextToken**
  - Characters. The token for the next page of results[optional]
- **DryRun**
  - Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
- **simplify**
  - Logical. Whether to simplify the result and handle `nextToken` in the response[optional]
- **others**
  - Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**
  - Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  Character. The region of the AWS service.

Value

A list object or a character vector

TransitGatewayAttachmentIds

One or more IDs of the transit gateway peering attachments.

Filter

One or more filters. The possible values are:

- transit-gateway-attachment-id - The ID of the transit gateway attachment.
- local-owner-id - The ID of your AWS account.
- remote-owner-id - The ID of the AWS account in the remote Region that owns the transit gateway.
- state - The state of the peering attachment. Valid values are available | deleted | deleting | failed | failing | initiatingRequest | modifying | pendingAcceptance | pending | rollingBack | rejected | rejecting).
- tag: - The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA for the filter value.
- tag-key - The key of a tag assigned to the resource. Use this filter to find all resources that have a tag with a specific key, regardless of the tag value.
- transit-gateway-id - The ID of the transit gateway.

MaxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
Describe Transit Gateway Route Tables

Description

Describes one or more transit gateway route tables. By default, all transit gateway route tables are described. Alternatively, you can filter the results.

Usage

```r
e2_describe_transit_gateway_route_tables(
  TransitGatewayRouteTableIds = NULL,
  Filter = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **TransitGatewayRouteTableIds**
  List. The IDs of the transit gateway route tables.[optional]

- **Filter**
  Named list where the name is the filter name and the value is the filter's value.
  The value can be a vector or a list object (see below)[optional]

- **MaxResults**
  Integer. The maximum number of results to return with a single call.[optional]

- **NextToken**
  Characters. The token for the next page of results[optional]

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]

- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  Character. The region of the AWS service.
Value

A list object or a character vector

TransitGatewayRouteTableIds

The IDs of the transit gateway route tables.

Filter

One or more filters. The possible values are:

- default-association-route-table - Indicates whether this is the default association route table for the transit gateway (true \| false).
- default-propagation-route-table - Indicates whether this is the default propagation route table for the transit gateway (true \| false).
- state - The state of the route table (available \| deleting \| deleted \| pending).
- transit-gateway-id - The ID of the transit gateway.
- transit-gateway-route-table-id - The ID of the transit gateway route table.

MaxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description

Describes one or more VPC attachments. By default, all VPC attachments are described. Alternatively, you can filter the results.
Usage

ec2_describe_transit_gateway_vpc_attachments(
    TransitGatewayAttachmentIds = NULL,
    Filter = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

TransitGatewayAttachmentIds
List. The IDs of the attachments.[optional]

Filter
Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below).[optional]

MaxResults
Integer. The maximum number of results to return with a single call.[optional]

NextToken
Characters. The token for the next page of results[optional]

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value

A list object or a character vector

TransitGatewayAttachmentIds

The IDs of the attachments.
Filter

One or more filters. The possible values are:

- **state** - The state of the attachment. Valid values are available | deleted | deleting | failed | failing | initiatingRequest | modifying | pendingAcceptance | pending | rollingBack | rejected | rejecting.

- **transit-gateway-attachment-id** - The ID of the attachment.

- **transit-gateway-id** - The ID of the transit gateway.

- **vpc-id** - The ID of the VPC.

MaxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

```
ec2_describe_volumes(  
  Filter = NULL,  
  VolumeId = NULL,  
  DryRun = NULL,  
  MaxResults = NULL,  
  NextToken = NULL,  
  simplify = TRUE,  
  others = list(),  
  print_on_error = aws_get_print_on_error(),  
  retry_time = aws_get_retry_time(),  
  network_timeout = aws_get_network_timeout(),  
  region = aws_get_region()  
)
```
Arguments

Filter  Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]
VolumeId  List. The volume IDs.[optional]
DryRun  Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
MaxResults  Integer. The maximum number of volume results returned by DescribeVolumes in paginated output.[optional]
NextToken  Characters. The token for the next page of results[optional]
simplify  Logical. Whether to simplify the result and handle nextToken in the response[optional]
others  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error  Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region  Character. The region of the AWS service.

Value

A list object or a character vector

Filter

The filters.

• attachment.attach-time - The time stamp when the attachment initiated.
• attachment.delete-on-termination - Whether the volume is deleted on instance termination.
• attachment.device - The device name specified in the block device mapping (for example, /dev/sda1).
• attachment.instance-id - The ID of the instance the volume is attached to.
• attachment.status - The attachment state (attaching \ attached \ detaching).
• availability-zone - The Availability Zone in which the volume was created.
• create-time - The time stamp when the volume was created.
• encrypted - Indicates whether the volume is encrypted (true \ false)
• multi-attach-enabled - Indicates whether the volume is enabled for Multi-Attach (true \ false)
• fast-restored - Indicates whether the volume was created from a snapshot that is enabled for fast snapshot restore (true \ false).
• size - The size of the volume, in GiB.
• snapshot-id - The snapshot from which the volume was created.
• status - The state of the volume (creating | available | in-use | deleting | deleted | error).
• tag: - The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA for the filter value.
• tag-key - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.
• volume-id - The volume ID.
• volume-type - The Amazon EBS volume type (gp2 | gp3 | io1 | io2 | st1 | sc1 | standard)

VolumeId

The volume IDs.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

MaxResults

The maximum number of volume results returned by DescribeVolumes in paginated output. When this parameter is used, DescribeVolumes only returns MaxResults results in a single page along with a NextToken response element. The remaining results of the initial request can be seen by sending another DescribeVolumes request with the returned NextToken value. This value can be between 5 and 500; if MaxResults is given a value larger than 500, only 500 results are returned. If this parameter is not used, then DescribeVolumes returns all results. You cannot specify this parameter and the volume IDs parameter in the same request.
Usage

e2_describe_volumes_modifications(
    DryRun = NULL,
    VolumeId = NULL,
    Filter = NULL,
    NextToken = NULL,
    MaxResults = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

 señ DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

* VolumeId List. The IDs of the volumes.

* Filter Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below).

* NextToken Characters. The token for the next page of results.

* MaxResults Integer. The maximum number of results (up to a limit of 500) to be returned in a paginated request.

* simplify Logical. Whether to simplify the result and handle nextToken in the response.

* others Named list. The parameters that are not included in the function parameters and need to be added into the request.

* print_on_error Logical. Whether to show an error message when a network error occurs.

* retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

* network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

* region Character. The region of the AWS service.

Value

A list object or a character vector

DryRun Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
VolumeId

The IDs of the volumes.

Filter

The filters.

- modification-state - The current modification state (modifying \| optimizing \| completed \| failed).
- original-iops - The original IOPS rate of the volume.
- original-size - The original size of the volume, in GiB.
- original-volume-type - The original volume type of the volume (standard \| io1 \| io2 \| gp2 \| sc1 \| st1).
- originalMultiAttachEnabled - Indicates whether Multi-Attach support was enabled (true \| false).
- start-time - The modification start time.
- target-iops - The target IOPS rate of the volume.
- target-size - The target size of the volume, in GiB.
- target-volume-type - The target volume type of the volume (standard \| io1 \| io2 \| gp2 \| sc1 \| st1).
- targetMultiAttachEnabled - Indicates whether Multi-Attach support is to be enabled (true \| false).
- volume-id - The ID of the volume.

MaxResults

The maximum number of results (up to a limit of 500) to be returned in a paginated request.

---

**Description**

Describe Volume Attribute

**Usage**

```python
e2_describe_volume_attribute(
    Attribute,
    VolumeId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
)```
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

Attribute           Character. The attribute of the volume. This parameter is required.
VolumeId            Character. The ID of the volume.
DryRun              Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
simplify            Logical. Whether to simplify the result and handle nextToken in the response [optional]
others              Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]
print_on_error      Logical. Whether to show an error message when a network error occurs.
retry_time          Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout     Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region              Character. The region of the AWS service.

Value

A list object or a character vector

Attribute

The attribute of the volume. This parameter is required.

VolumeId

The ID of the volume.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
ec2_describe_volume_status

Describe Volume Status

Description

Describe Volume Status

Usage

c2.describe_volume_status(
  Filter = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  VolumeId = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws.get_print_on_error(),
  retry_time = aws.get_retry_time(),
  network_timeout = aws.get_network_timeout(),
  region = aws.get_region()
)

Arguments

Filter Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]
MaxResults Integer. The maximum number of volume results returned by DescribeVolumeStatus in paginated output.[optional]
NextToken Characters. The token for the next page of results[optional]
VolumeId List. The IDs of the volumes. Default: Describes all your volumes. [optional]
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.
Value

A list object or a character vector

Filter

The filters.

- action.code - The action code for the event (for example, enable-volume-io).
- action.description - A description of the action.
- action.event-id - The event ID associated with the action.
- availability-zone - The Availability Zone of the instance.
- event.description - A description of the event.
- event.event-id - The event ID.
- event.not-after - The latest end time for the event.
- event.not-before - The earliest start time for the event.
- volume-status.status - The status of the volume (ok | impaired | warning | insufficient-data).

MaxResults

The maximum number of volume results returned by DescribeVolumeStatus in paginated output. When this parameter is used, the request only returns MaxResults results in a single page along with a NextToken response element. The remaining results of the initial request can be seen by sending another request with the returned NextToken value. This value can be between 5 and 1,000; if MaxResults is given a value larger than 1,000, only 1,000 results are returned. If this parameter is not used, then DescribeVolumeStatus returns all results. You cannot specify this parameter and the volume IDs parameter in the same request.

VolumeId

The IDs of the volumes.

Default: Describes all your volumes.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
ec2_describe_vpcs

Description

Describes one or more of your VPCs.

Usage

ec2_describe_vpcs(
  Filter = NULL,
  VpcId = NULL,
  DryRun = NULL,
  NextToken = NULL,
  MaxResults = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

Filter  Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]
VpcId   List. One or more VPC IDs. Default: Describes all your VPCs. [optional]
DryRun  Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
NextToken  Characters. The token for the next page of results[optional]
MaxResults  Integer. The maximum number of results to return with a single call.[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region  Character. The region of the AWS service.
Value

A list object or a character vector

Filter

One or more filters.

- **cidr**: The primary IPv4 CIDR block of the VPC. The CIDR block you specify must exactly match the VPC’s CIDR block for information to be returned for the VPC. Must contain the slash followed by one or two digits (for example, /28).
- **cidr-block-association.cidr-block**: An IPv4 CIDR block associated with the VPC.
- **cidr-block-association.association-id**: The association ID for an IPv4 CIDR block associated with the VPC.
- **cidr-block-association.state**: The state of an IPv4 CIDR block associated with the VPC.
- **dhcp-options-id**: The ID of a set of DHCP options.
- **ipv6-cidr-block-association.ipv6-cidr-block**: An IPv6 CIDR block associated with the VPC.
- **ipv6-cidr-block-association.ipv6-pool**: The ID of the IPv6 address pool from which the IPv6 CIDR block is allocated.
- **ipv6-cidr-block-association.association-id**: The association ID for an IPv6 CIDR block associated with the VPC.
- **ipv6-cidr-block-association.state**: The state of an IPv6 CIDR block associated with the VPC.
- **isDefault**: Indicates whether the VPC is the default VPC.
- **owner-id**: The ID of the AWS account that owns the VPC.
- **state**: The state of the VPC (pending | available).
- **tag**: The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA for the filter value.
- **tag-key**: The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.
- **vpc-id**: The ID of the VPC.

VpcId

One or more VPC IDs.

Default: Describes all your VPCs.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
MaxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

Arguments

Attribute Character. The VPC attribute.
VpcId Character. The ID of the VPC.
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response [optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.
ec2_describe_vpc_classic_link

Value
A list object or a character vector

Attribute
The VPC attribute.

VpcId
The ID of the VPC.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_describe_vpc_classic_link

Describe Vpc Classic Link

Description
Describes the ClassicLink status of one or more VPCs.

Usage
ec2_describe_vpc_classic_link(
  Filter = NULL,
  DryRun = NULL,
  VpcId = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments
Filter          Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]
DryRun          Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]
VpcId           List. One or more VPCs for which you want to describe the ClassicLink status.[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value
A list object or a character vector

Filter
One or more filters.

- **is-classic-link-enabled** - Whether the VPC is enabled for ClassicLink (true \| false).
- **tag:\key** - The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA for the filter value.
- **tag-key** - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

VpcId
One or more VPCs for which you want to describe the ClassicLink status.
ec2_describe_vpc_classic_link_dns_support

Describe Vpc Classic Link Dns Support

Description

Describes the ClassicLink DNS support status of one or more VPCs. If enabled, the DNS hostname of a linked EC2-Classic instance resolves to its private IP address when addressed from an instance in the VPC to which it's linked. Similarly, the DNS hostname of an instance in a VPC resolves to its private IP address when addressed from a linked EC2-Classic instance. For more information, see ClassicLink in the Amazon Elastic Compute Cloud User Guide.

Usage

ec2_describe_vpc_classic_link_dns_support(
    MaxResults = NULL,
    NextToken = NULL,
    VpcIds = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

MaxResults  Integer. The maximum number of results to return with a single call.[optional]
NextToken   Characters. The token for the next page of results[optional]
VpcIds      List. One or more VPC IDs.[optional]
simplify    Logical. Whether to simplify the result and handle nextToken in the response[optional]
others      Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time   Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region      Character. The region of the AWS service.

Value

A list object or a character vector
MaxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

VpcIds

One or more VPC IDs.

Arguments

DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request,[optional]
VpcEndpointId List. One or more endpoint IDs,[optional]
Filter Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]
MaxResults Integer. The maximum number of items to return for this request,[optional]
NextToken Characters. The token for the next page of results[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  Character. The region of the AWS service.

Value

A list object or a character vector

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

VpcEndpointId

One or more endpoint IDs.

Filter

One or more filters.

- service-name - The name of the service.
- vpc-id - The ID of the VPC in which the endpoint resides.
- vpc-endpoint-id - The ID of the endpoint.
- vpc-endpoint-state - The state of the endpoint (pendingAcceptance \| pending \| available \| deleting \| deleted \| rejected \| failed).
- vpc-endpoint-type - The type of VPC endpoint (Interface \| Gateway \| GatewayLoadBalancer).
- tag:\ - The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA for the filter value.
- tag-key - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.

MaxResults

The maximum number of items to return for this request. The request returns a token that you can specify in a subsequent call to get the next set of results.

Constraint: If the value is greater than 1,000, we return only 1,000 items.
Describe Vpc Endpoint Connections

Description

Describes the VPC endpoint connections to your VPC endpoint services, including any endpoints that are pending your acceptance.

Usage

```r
e2_describe_vpc_endpoint_connections(
  DryRun = NULL,
  Filter = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **DryRun** Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **Filter** Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below) [optional]
- **MaxResults** Integer. The maximum number of results to return for the request in a single page. [optional]
- **NextToken** Characters. The token for the next page of results [optional]
- **simplify** Logical. Whether to simplify the result and handle `nextToken` in the response [optional]
- **others** Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]
- **print_on_error** Logical. Whether to show an error message when a network error occurs.
- **retry_time** Integer. Number of retries for a REST request when encountering the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout** Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.
- **region** Character. The region of the AWS service.
ec2_describe_vpc_endpoint_connection_notifications

Description

Describes the connection notifications for VPC endpoints and VPC endpoint services.

Usage

```r
ec2_describe_vpc_endpoint_connection_notifications(
  DryRun = NULL,
  ConnectionNotificationId = NULL,
  Filter = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
)```
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

**DryRun**
Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]

**ConnectionNotificationId**
Character. The ID of the notification. [optional]

**Filter**
Named list where the name is the filter name and the value is the filter's value.
The value can be a vector or a list object (see below). [optional]

**MaxResults**
Integer. The maximum number of results to return in a single call. [optional]

**NextToken**
Characters. The token for the next page of results. [optional]

**simplify**
Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]

**others**
Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]

**print_on_error**
Logical. Whether to show an error message when a network error occurs.

**retry_time**
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

**network_timeout**
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

**region**
Character. The region of the AWS service.

Value
A list object or a character vector

**DryRun**
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

**ConnectionNotificationId**
The ID of the notification.

**Filter**
One or more filters.

- `connection-notification-arn` - The ARN of the SNS topic for the notification.
- `connection-notification-id` - The ID of the notification.
connection-notification-state - The state of the notification (Enabled | Disabled).
connection-notification-type - The type of notification (Topic).
service-id - The ID of the endpoint service.
vpc-endpoint-id - The ID of the VPC endpoint.

MaxResults
The maximum number of results to return in a single call. To retrieve the remaining results, make another request with the returned NextToken value.

ec2_describe_vpc_endpoint_services

Describe Vpc Endpoint Services

Usage

ec2_describe_vpc_endpoint_services(
    DryRun = NULL,
    ServiceName = NULL,
    Filter = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
ServiceName List. One or more service names.[optional]
Filter Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]
MaxResults Integer. The maximum number of items to return for this request.[optional]
NextToken Characters. The token for the next page of results[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
ec2_describe_vpc_endpoint_services

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value
A list object or a character vector

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ServiceName
One or more service names.

Filter
One or more filters.

- service-name - The name of the service.
- service-type - The type of service (Interface | Gateway).
- tag:
  - The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA for the filter value.
- tag-key - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.

MaxResults
The maximum number of items to return for this request. The request returns a token that you can specify in a subsequent call to get the next set of results.

Constraint: If the value is greater than 1,000, we return only 1,000 items.
ec2_describe_vpc_endpoint_service_configurations

Describe Vpc Endpoint Service Configurations

Description

Describes the VPC endpoint service configurations in your account (your services).

Usage

```r
ec2_describe_vpc_endpoint_service_configurations(
  DryRun = NULL,
  ServiceId = NULL,
  Filter = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
- **ServiceId**: List. The IDs of one or more services.[optional]
- **Filter**: Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below).[optional]
- **MaxResults**: Integer. The maximum number of results to return for the request in a single page.[optional]
- **NextToken**: Characters. The token for the next page of results.[optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response.[optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request.[optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.
Value

A list object or a character vector

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ServiceId

The IDs of one or more services.

Filter

One or more filters.

- service-name - The name of the service.
- service-id - The ID of the service.
- service-state - The state of the service (Pending | Available | Deleting | Deleted | Failed).
- tag: - The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA for the filter value.
- tag-key - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.

MaxResults

The maximum number of results to return for the request in a single page. The remaining results of the initial request can be seen by sending another request with the returned NextToken value. This value can be between 5 and 1,000; if MaxResults is given a value larger than 1,000, only 1,000 results are returned.

---

**Description**

Describes the principals (service consumers) that are permitted to discover your VPC endpoint service.
Usage

```r
ec2_describe_vpc_endpoint_service_permissions(
  ServiceId,
  DryRun = NULL,
  Filter = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **ServiceId**: Character. The ID of the service.
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request. (optional)
- **Filter**: Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below) (optional)
- **MaxResults**: Integer. The maximum number of results to return for the request in a single page. (optional)
- **NextToken**: Characters. The token for the next page of results. (optional)
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response. (optional)
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request. (optional)
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

**ServiceId**

The ID of the service.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Filter

One or more filters.

- principal - The ARN of the principal.
- principal-type - The principal type (All \| Service \| OrganizationUnit \| Account \| User \| Role).

MaxResults

The maximum number of results to return for the request in a single page. The remaining results of the initial request can be seen by sending another request with the returned NextToken value. This value can be between 5 and 1,000; if MaxResults is given a value larger than 1,000, only 1,000 results are returned.

ec2_describe_vpc_peering_connections

Describe Vpc Peering Connections

Description

Describes one or more of your VPC peering connections.

Usage

ec2_describe_vpc_peering_connections(
  Filter = NULL,
  DryRun = NULL,
  VpcPeeringConnectionId = NULL,
  NextToken = NULL,
  MaxResults = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
ec2_describe_vpc_peering_connections

Arguments

Filter
Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]

VpcPeeringConnectionId
List. One or more VPC peering connection IDs. Default: Describes all your VPC peering connections. [optional]

NextToken
Characters. The token for the next page of results[optional]

MaxResults
Integer. The maximum number of results to return with a single call.[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value
A list object or a character vector

Filter
One or more filters.

• accepter-vpc-info.cidr-block - The IPv4 CIDR block of the accepter VPC.
• accepter-vpc-info.owner-id - The AWS account ID of the owner of the accepter VPC.
• accepter-vpc-info.vpc-id - The ID of the accepter VPC.
• expiration-time - The expiration date and time for the VPC peering connection.
• requester-vpc-info.cidr-block - The IPv4 CIDR block of the requester’s VPC.
• requester-vpc-info.owner-id - The AWS account ID of the owner of the requester VPC.
• requester-vpc-info.vpc-id - The ID of the requester VPC.
• status-code - The status of the VPC peering connection (pending-acceptance \| failed \| expired \| provisioning \| active \| deleting \| deleted \| rejected).
• status-message - A message that provides more information about the status of the VPC peering connection, if applicable.
- **tag**: The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key `Owner` and the value `TeamA`, specify `tag:Owner` for the filter name and `TeamA` for the filter value.
- **tag-key**: The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.
- **vpc-peering-connection-id**: The ID of the VPC peering connection.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

**VpcPeeringConnectionId**

One or more VPC peering connection IDs.

Default: Describes all your VPC peering connections.

**MaxResults**

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned `nextToken` value.

---

**ec2_describe_vpn_connections (Describe Vpn Connections)**

**Description**

Describe Vpn Connections

**Usage**

```r
ec2_describe_vpn_connections(
  Filter = NULL,
  VpnConnectionId = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)```


Arguments

Filter
Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]

VpnConnectionId
List. One or more VPN connection IDs. Default: Describes your VPN connections. [optional]

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request,[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value
A list object or a character vector

Filter
One or more filters.

- customer-gateway-configuration - The configuration information for the customer gateway.
- customer-gateway-id - The ID of a customer gateway associated with the VPN connection.
- state - The state of the VPN connection (pending | available | deleting | deleted).
- option.static-routes-only - Indicates whether the connection has static routes only. Used for devices that do not support Border Gateway Protocol (BGP).
- route.destination-cidr-block - The destination CIDR block. This corresponds to the subnet used in a customer data center.
- bgp-asn - The BGP Autonomous System Number (ASN) associated with a BGP device.
- tag:\ - The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA for the filter value.
- tag-key - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.
- type - The type of VPN connection. Currently the only supported type is ipsec.1.
- **vpn-connection-id** - The ID of the VPN connection.
- **vpn-gateway-id** - The ID of a virtual private gateway associated with the VPN connection.
- **transit-gateway-id** - The ID of a transit gateway associated with the VPN connection.

### VpnConnectionId

One or more VPN connection IDs.

Default: Describes your VPN connections.

### DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is **DryRunOperation**. Otherwise, it is **UnauthorizedOperation**.

---

**Usage**

```
ec2_describe_vpn_gateways(
    Filter = NULL,
    VpnGatewayId = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

### Arguments

- **Filter**
  Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]

- **VpnGatewayId**
  List. One or more virtual private gateway IDs. Default: Describes all your virtual private gateways. [optional]

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]

- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response[optional]
others  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.

Value
A list object or a character vector

Filter
One or more filters.

• amazon-side-asn - The Autonomous System Number (ASN) for the Amazon side of the gateway.

• attachment.state - The current state of the attachment between the gateway and the VPC (attaching | attached | detaching | detached).

• attachment.vpc-id - The ID of an attached VPC.

• availability-zone - The Availability Zone for the virtual private gateway (if applicable).

• state - The state of the virtual private gateway (pending | available | deleting | deleted).

• tag: - The key/value combination of a tag assigned to the resource. Use the tag key in the filter name and the tag value as the filter value. For example, to find all resources that have a tag with the key Owner and the value TeamA, specify tag:Owner for the filter name and TeamA for the filter value.

• tag-key - The key of a tag assigned to the resource. Use this filter to find all resources assigned a tag with a specific key, regardless of the tag value.

• type - The type of virtual private gateway. Currently the only supported type is ipsec.1.

• vpn-gateway-id - The ID of the virtual private gateway.

VpnGatewayId
One or more virtual private gateway IDs.
Default: Describes all your virtual private gateways.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
Detach Classic Link Vpc

Description
Unlinks (detaches) a linked EC2-Classic instance from a VPC. After the instance has been unlinked, the VPC security groups are no longer associated with it. An instance is automatically unlinked from a VPC when it’s stopped.

Usage
```r
ec2_detach_classic_link_vpc(
  InstanceId,
  VpcId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments
- **InstanceId** Character. The ID of the instance to unlink from the VPC.
- **VpcId** Character. The ID of the VPC to which the instance is linked.
- **DryRun** Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **simplify** Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]
- **others** Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]
- **print_on_error** Logical. Whether to show an error message when a network error occurs.
- **retry_time** Integer. Number of retries for a REST request when encountering a network issue. If the request has been sent `retry_time` times but still not able to get the response, an error will be thrown.
- **network_timeout** Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region** Character. The region of the AWS service.

Value
A list object or a character vector
InstanceId

The ID of the instance to unlink from the VPC.

VpcId

The ID of the VPC to which the instance is linked.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description

Detaches an internet gateway from a VPC, disabling connectivity between the internet and the VPC. The VPC must not contain any running instances with Elastic IP addresses or public IPv4 addresses.

Usage

```r
ec2_detach_internet_gateway(
  InternetGatewayId,
  VpcId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **InternetGatewayId**
  Character. The ID of the internet gateway.
- **VpcId**
  Character. The ID of the VPC.
- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
**ec2_detach_network_interface**

*Detach Network Interface*

**Description**

Detaches a network interface from an instance.

**Usage**

```r
ec2_detach_network_interface(
  AttachmentId,
  DryRun = NULL,
  Force = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**print_on_error**

Logical. Whether to show an error message when a network error occurs.

**retry_time**

Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

**network_timeout**

Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

**region**

Character. The region of the AWS service.

**Value**

A list object or a character vector

**InternetGatewayId**

The ID of the internet gateway.

**VpcId**

The ID of the VPC.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AttachmentId</td>
<td>Character</td>
<td>The ID of the attachment.</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical</td>
<td>Checks whether you have the required permissions for the action, without actually making the request.</td>
</tr>
<tr>
<td>Force</td>
<td>Logical</td>
<td>Specifies whether to force a detachment. [optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical</td>
<td>Whether to simplify the result and handle nextToken in the response [optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list</td>
<td>The parameters that are not included in the function parameters and need to be added into the request [optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical</td>
<td>Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer</td>
<td>Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric</td>
<td>Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character</td>
<td>The region of the AWS service.</td>
</tr>
</tbody>
</table>

Value

A list object or a character vector

AttachmentId

The ID of the attachment.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Force

Specifies whether to force a detachment.

- Use the Force parameter only as a last resort to detach a network interface from a failed instance.
- If you use the Force parameter to detach a network interface, you might not be able to attach a different network interface to the same index on the instance without first stopping and starting the instance.
- If you force the detachment of a network interface, the instance metadata might not get updated. This means that the attributes associated with the detached network interface might still be visible. The instance metadata will get updated when you stop and start the instance.
ec2_detach_volume

Description

Detach Volume

Usage

ec2_detach_volume(
  VolumeId,
  Device = NULL,
  Force = NULL,
  InstanceId = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VolumeId</td>
<td>Character. The ID of the volume.</td>
</tr>
<tr>
<td>Device</td>
<td>Character. The device name.[optional]</td>
</tr>
<tr>
<td>Force</td>
<td>Logical. Forces detachment if the previous detachment attempt did not occur cleanly (for example, logging into an instance, unmounting the volume, and detaching normally)...[optional]</td>
</tr>
<tr>
<td>InstanceId</td>
<td>Character. The ID of the instance.[optional]</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response[optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>
**Value**

A list object or a character vector

**VolumeId**

The ID of the volume.

**Device**

The device name.

**Force**

Forces detachment if the previous detachment attempt did not occur cleanly (for example, logging into an instance, unmounting the volume, and detaching normally). This option can lead to data loss or a corrupted file system. Use this option only as a last resort to detach a volume from a failed instance. The instance won't have an opportunity to flush file system caches or file system metadata. If you use this option, you must perform file system check and repair procedures.

**InstanceId**

The ID of the instance. If you are detaching a Multi-Attach enabled volume, you must specify an instance ID.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

```r
ec2_detach_vpn_gateway

Detach Vpn Gateway

Description

Detach Vpn Gateway

Usage

ec2_detach_vpn_gateway(
  VpcId,
  VpnGatewayId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
)```
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

VpcId    Character. The ID of the VPC.
VpnGatewayId Character. The ID of the virtual private gateway.
DryRun    Logical. Checks whether you have the required permissions for the action, without actually making the request, ...[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region    Character. The region of the AWS service.

Value

A list object or a character vector

VpcId

The ID of the VPC.

VpnGatewayId

The ID of the virtual private gateway.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
Description

Disable Ebs Encryption By Default

Usage

```r
ec2_disable_ebs_encryption_by_default(
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`. 
Description

Disables fast snapshot restores for the specified snapshots in the specified Availability Zones.

Usage

```r
ec2_disable_fast_snapshot_restores(
    AvailabilityZone,
    SourceSnapshotId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **AvailabilityZone**
  List. One or more Availability Zones. For example, us-east-2a.

- **SourceSnapshotId**
  List. The IDs of one or more snapshots. For example, snap-1234567890abcdef0.

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]

- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encountering a network issue. If the request has been sent `retry_time` times but still not able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.

- **region**
  Character. The region of the AWS service.

Value

A list object or a character vector
AvailabilityZone

One or more Availability Zones. For example, us-east-2a.

SourceSnapshotId

The IDs of one or more snapshots. For example, snap-1234567890abcdef0.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description

Disables the specified resource attachment from propagating routes to the specified propagation route table.

Usage

ec2_disable_transit_gateway_route_table_propagation(
   TransitGatewayRouteTableId,
   TransitGatewayAttachmentId,
   DryRun = NULL,
   simplify = TRUE,
   others = list(),
   print_on_error = aws_get_print_on_error(),
   retry_time = aws_get_retry_time(),
   network_timeout = aws_get_network_timeout(),
   region = aws_get_region()
)

Arguments

TransitGatewayRouteTableId
   Character. The ID of the propagation route table.

TransitGatewayAttachmentId
   Character. The ID of the attachment.

DryRun
   Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

simplify
   Logical. Whether to simplify the result and handle nextToken in the response[optional]
ec2_disable_vgw_route propagation

Disables a virtual private gateway (VGW) from propagating routes to a specified route table of a VPC.

Usage

```
ec2_disable_vgw_route_propagation(
    GatewayId,
    RouteTableId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
```

Value

A list object or a character vector

**TransitGatewayRouteTableId**

The ID of the propagation route table.

**TransitGatewayAttachmentId**

The ID of the attachment.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

others  Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

GatewayId Character. The ID of the virtual private gateway.
RouteTableId Character. The ID of the route table.
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request,[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response,[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request,[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value

A list object or a character vector

GatewayId

The ID of the virtual private gateway.

RouteTableId

The ID of the route table.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
ec2_disable_vpc_classic_link

Disable Vpc Classic Link

Description
Disables ClassicLink for a VPC. You cannot disable ClassicLink for a VPC that has EC2-Classic instances linked to it.

Usage
ec2_disable_vpc_classic_link(
    VpcId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

VpcId Character. The ID of the VPC.
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value

A list object or a character vector

VpcId

The ID of the VPC.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

```r
ec2_disable_vpc_classic_link_dns_support

Description

Disable Vpc Classic Link Dns Support

Usage

```r
ec2_disable_vpc_classic_link_dns_support(
  VpcId = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **VpcId**: Character. The ID of the VPC.[optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response[optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

**VpcId**

The ID of the VPC.
Disassociate Address

Description
Disassociate Address

Usage
ec2_disassociate_address(
    AssociationId = NULL,
    PublicIp = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AssociationId</td>
<td>Character. [EC2-VPC] The association ID. Required for EC2-VPC. [optional]</td>
</tr>
<tr>
<td>PublicIp</td>
<td>Character. [EC2-Classic] The Elastic IP address. Required for EC2-Classic. [optional]</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response. [optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not able to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>

Value
A list object or a character vector
AssociationId

[EC2-VPC] The association ID. Required for EC2-VPC.

PublicIp


DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_disassociate_client_vpn_target_network

Disassociate Client Vpn Target Network

Description

Disassociate Client Vpn Target Network

Usage

ec2_disassociate_client_vpn_target_network(
    ClientVpnEndpointId,
    AssociationId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

ClientVpnEndpointId

Character. The ID of the Client VPN endpoint from which to disassociate the target network.

AssociationId

Character. The ID of the target network association.

DryRun

Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]

simplify

Logical. Whether to simplify the result and handle nextToken in the response[optional]

others

Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value

A list object or a character vector

ClientVpnEndpointId

The ID of the Client VPN endpoint from which to disassociate the target network.

AssociationId

The ID of the target network association.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

dec2_disassociate_enclave_certificate_iam_role

Description

Disassociates an IAM role from an AWS Certificate Manager (ACM) certificate. Disassociating an IAM role from an ACM certificate removes the Amazon S3 object that contains the certificate, certificate chain, and encrypted private key from the Amazon S3 bucket. It also revokes the IAM role's permission to use the AWS Key Management Service (KMS) customer master key (CMK) used to encrypt the private key. This effectively revokes the role's permission to use the certificate.

Usage

dec2_disassociate_enclave_certificate_iam_role(
   CertificateArn = NULL,
   RoleArn = NULL,
   DryRun = NULL,
   simplify = TRUE,
   others = list(),
)
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

CertificateArn  Character. The ARN of the ACM certificate from which to disassociate the IAM role.[optional]
RoleArn  Character. The ARN of the IAM role to disassociate.[optional]
DryRun  Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
simplify  Logical. Whether to simplify the result and handle nextToken in the response[optional]
others  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error  Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region  Character. The region of the AWS service.

Value

A list object or a character vector

CertificateArn

The ARN of the ACM certificate from which to disassociate the IAM role.

RoleArn

The ARN of the IAM role to disassociate.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
**ec2_disassociate_iam_instance_profile**

*Disassociate Iam Instance Profile*

---

**Description**

Disassociate Iam Instance Profile

**Usage**

```r
ec2_disassociate_iam_instance_profile(
  AssociationId,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- **AssociationId**
  Character. The ID of the IAM instance profile association.

- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response [optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  Character. The region of the AWS service.

**Value**

A list object or a character vector

**AssociationId**

The ID of the IAM instance profile association.
ec2_disassociate_route_table

Disassociate Route Table

Description

Disassociate Route Table

Usage

ec2_disassociate_route_table(
  AssociationId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

AssociationId  Character. The association ID representing the current association between the route table and subnet or gateway.

DryRun        Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

simplify      Logical. Whether to simplify the result and handle nextToken in the response[optional]

others        Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time    Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region        Character. The region of the AWS service.

Value

A list object or a character vector

AssociationId

The association ID representing the current association between the route table and subnet or gateway.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description

Disassociates a CIDR block from a subnet. Currently, you can disassociate an IPv6 CIDR block only. You must detach or delete all gateways and resources that are associated with the CIDR block before you can disassociate it.

Usage

```r
ec2_disassociate_subnet_cidr_block(
  AssociationId,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **AssociationId**: Character. The association ID for the CIDR block.
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response [optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector
AssociationId

The association ID for the CIDR block.

---

**Description**

Disassociates the specified subnets from the transit gateway multicast domain.

**Usage**

```r
ec2_disassociate_transit_gateway_multicast_domain(
  TransitGatewayMulticastDomainId = NULL,
  TransitGatewayAttachmentId = NULL,
  SubnetIds = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- **TransitGatewayMulticastDomainId**
  - Character. The ID of the transit gateway multicast domain.[optional]
- **TransitGatewayAttachmentId**
  - Character. The ID of the attachment.[optional]
- **SubnetIds**
  - List. The IDs of the subnets:[optional]
- **DryRun**
  - Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
- **simplify**
  - Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others**
  - Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**
  - Logical. Whether to show an error message when a network error occurs.
- **retry_time**
  - Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- **network_timeout**
  - Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**
  - Character. The region of the AWS service.
Disassociate Transit Gateway Route Table

Description

Disassociates a resource attachment from a transit gateway route table.

Usage

```r
ec2_disassociate_transit_gateway_route_table(
    TransitGatewayRouteTableId,
    TransitGatewayAttachmentId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```
Arguments

TransitGatewayRouteTableId
Character. The ID of the transit gateway route table.

TransitGatewayAttachmentId
Character. The ID of the attachment.

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value

A list object or a character vector

TransitGatewayRouteTableId

The ID of the transit gateway route table.

TransitGatewayAttachmentId

The ID of the attachment.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
ec2_disassociate_vpc_cidr_block

Disassociate Vpc Cidr Block

Description

Disassociate Vpc Cidr Block

Usage

ec2_disassociate_vpc_cidr_block(
  AssociationId,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

AssociationId  Character. The association ID for the CIDR block.
simplify       Logical. Whether to simplify the result and handle nextToken in the response[optional]
others         Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error  Logical. Whether to show an error message when a network error occurs.
retry_time      Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region         Character. The region of the AWS service.

Value

A list object or a character vector

AssociationId

The association ID for the CIDR block.
ec2_enable_ebs_encryption_by_default

Enable Ebs Encryption By Default

Description
Enable Ebs Encryption By Default

Usage
ec2_enable_ebs_encryption_by_default(
DryRun = NULL,
simplify = TRUE,
others = list(),
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response [optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value
A list object or a character vector

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
ec2_enable_fast_snapshot_restores

Enable Fast Snapshot Restores

Description

Enable Fast Snapshot Restores

Usage

ec2_enable_fast_snapshot_restores(
    AvailabilityZone,
    SourceSnapshotId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

AvailabilityZone
    List. One or more Availability Zones. For example, us-east-2a.
SourceSnapshotId
    List. The IDs of one or more snapshots.
DryRun
    Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
simplify
    Logical. Whether to simplify the result and handle nextToken in the response[optional]
others
    Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error
    Logical. Whether to show an error message when a network error occurs.
retry_time
    Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout
    Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region
    Character. The region of the AWS service.

Value

A list object or a character vector
AvailabilityZone

One or more Availability Zones. For example, us-east-2a.

SourceSnapshotId

The IDs of one or more snapshots. For example, snap-1234567890abcdef0. You can specify a snapshot that was shared with you from another AWS account.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

dc2_enable_transit_gateway_route_table_propagation

Enable Transit Gateway Route Table Propagation

Description

Enables the specified attachment to propagate routes to the specified propagation route table.

Usage

```r
ec2_enable_transit_gateway_route_table_propagation(
  TransitGatewayRouteTableId,
  TransitGatewayAttachmentId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **TransitGatewayRouteTableId**
  Character. The ID of the propagation route table.

- **TransitGatewayAttachmentId**
  Character. The ID of the attachment.

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]

- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response[optional]
ec2_enable_vgw_route_propagation

A list object or a character vector

TransitGatewayRouteTableId

The ID of the propagation route table.

TransitGatewayAttachmentId

The ID of the attachment.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

ect2_enable_vgw_route_propagation

Enable Vgw Route Propagation

Description

Enables a virtual private gateway (VGW) to propagate routes to the specified route table of a VPC.

Usage

```r
c2_enable_vgw_route_propagation(
    GatewayId,
    RouteTableId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
```
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

GatewayId          Character. The ID of the virtual private gateway that is attached to a VPC.
RouteTableId       Character. The ID of the route table.
DryRun             Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
simplify           Logical. Whether to simplify the result and handle nextToken in the response [optional]
others             Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]
print_on_error      Logical. Whether to show an error message when a network error occurs.
retry_time          Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout     Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.
region              Character. The region of the AWS service.

Value

A list object or a character vector

GatewayId

The ID of the virtual private gateway that is attached to a VPC. The virtual private gateway must be attached to the same VPC that the routing tables are associated with.

RouteTableId

The ID of the route table. The routing table must be associated with the same VPC that the virtual private gateway is attached to.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
ec2_enable_volume_io

**Description**

Enables I/O operations for a volume that had I/O operations disabled because the data on the volume was potentially inconsistent.

**Usage**

```r
ec2_enable_volume_io(
  VolumeId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- **VolumeId** Character. The ID of the volume.
- **DryRun** Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
- **simplify** Logical. Whether to simplify the result and handle `nextToken` in the response[optional]
- **others** Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error** Logical. Whether to show an error message when a network error occurs.
- **retry_time** Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout** Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region** Character. The region of the AWS service.

**Value**

A list object or a character vector

**VolumeId**

The ID of the volume.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description

Enables a VPC for ClassicLink. You can then link EC2-Classic instances to your ClassicLink-enabled VPC to allow communication over private IP addresses. You cannot enable your VPC for ClassicLink if any of your VPC route tables have existing routes for address ranges within the 10.0.0.0/8 IP address range, excluding local routes for VPCs in the 10.0.0.0/16 and 10.1.0.0/16 IP address ranges. For more information, see ClassicLink in the Amazon Elastic Compute Cloud User Guide.

Usage

```r
ec2_enable_vpc_classic_link(
  VpcId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VpcId</td>
<td>Character. The ID of the VPC.</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action,</td>
</tr>
<tr>
<td></td>
<td>without actually making the request,...[optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response[optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters</td>
</tr>
<tr>
<td></td>
<td>and need to be added into the request[optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network</td>
</tr>
<tr>
<td></td>
<td>issue. If the request has been sent retry_time times but still not able to</td>
</tr>
<tr>
<td></td>
<td>get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can</td>
</tr>
<tr>
<td></td>
<td>not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>
**ec2_enable_vpc_classic_link_dns_support**

**Value**
A list object or a character vector

**VpcId**
The ID of the VPC.

**DryRun**
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is **DryRunOperation**. Otherwise, it is **UnauthorizedOperation**.

---

```
ec2_enable_vpc_classic_link_dns_support
   Enable Vpc Classic Link Dns Support
```

**Description**
Enable Vpc Classic Link Dns Support

**Usage**
```
ec2_enable_vpc_classic_link_dns_support(
    VpcId = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VpcId</td>
<td>Character. The ID of the VPC. [optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response. [optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>
Value
A list object or a character vector

VpcId
The ID of the VPC.

ec2_export_client_vpn_client_certificate_revocation_list
Export Client Vpn Client Certificate Revocation List

Description
Downloads the client certificate revocation list for the specified Client VPN endpoint.

Usage
ec2_export_client_vpn_client_certificate_revocation_list(
    ClientVpnEndpointId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments
ClientVpnEndpointId Character. The ID of the Client VPN endpoint.
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.
Value
A list object or a character vector

ClientVpnEndpointId
The ID of the Client VPN endpoint.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description
Downloads the contents of the Client VPN endpoint configuration file for the specified Client VPN endpoint. The Client VPN endpoint configuration file includes the Client VPN endpoint and certificate information clients need to establish a connection with the Client VPN endpoint.

Usage
ec2_export_client_vpn_client_configuration(
    ClientVpnEndpointId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments
ClientVpnEndpointId
    Character. The ID of the Client VPN endpoint.

DryRun
    Logical. Checks whether you have the required permissions for the action, without actually making the request, [optional]

simplify
    Logical. Whether to simplify the result and handle nextToken in the response [optional]

others
    Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]

print_on_error
    Logical. Whether to show an error message when a network error occurs.
**retry_time**  
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

**network_timeout**  
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

**region**  
Character. The region of the AWS service.

**Value**  
A list object or a character vector

**ClientVpnEndpointId**  
The ID of the Client VPN endpoint.

**DryRun**  
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**ec2_export_image**  
**Export Image**

**Description**  
Exports an Amazon Machine Image (AMI) to a VM file. For more information, see Exporting a VM Directory from an Amazon Machine Image (AMI) in the VM Import/Export User Guide.

**Usage**

```r
ec2_export_image(
  DiskImageFormat,
  ImageId,
  S3ExportLocation,
  ClientToken = NULL,
  Description = NULL,
  DryRun = NULL,
  RoleName = NULL,
  TagSpecification = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
**ec2_export_image**

**Arguments**

- **DiskImageFormat**
  Character. The disk image format.

- **ImageId**
  Character. The ID of the image.

- **S3ExportLocation**
  Object. Information about the destination Amazon S3 bucket.

- **ClientToken**
  Character. Token to enable idempotency for export image requests.[optional]

- **Description**
  Character. A description of the image being exported. The maximum length is 255 characters.[optional]

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]

- **RoleName**
  Character. The name of the role that grants VM Import/Export permission to export images to your Amazon S3 bucket...[optional]

- **TagSpecification**
  List. The tags to apply to the export image task during creation.[optional]

- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  Character. The region of the AWS service.

**Value**

A list object or a character vector

**DiskImageFormat**

The disk image format.

**ImageId**

The ID of the image.

**S3ExportLocation**

Information about the destination Amazon S3 bucket. The bucket must exist and grant WRITE and READ_ACP permissions to the AWS account vm-import-export\@amazon.com.
ClientToken

Token to enable idempotency for export image requests.

Description

A description of the image being exported. The maximum length is 255 characters.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

RoleName

The name of the role that grants VM Import/Export permission to export images to your Amazon S3 bucket. If this parameter is not specified, the default role is named 'vmimport'.

TagSpecification

The tags to apply to the export image task during creation.

---

```r
ec2_export_transit_gateway_routes

Export Transit Gateway Routes

Description

Export Transit Gateway Routes

Usage

ec2_export_transit_gateway_routes(
  TransitGatewayRouteTableId,  
  S3Bucket,  
  Filter = NULL,  
  DryRun = NULL,  
  simplify = TRUE,  
  others = list(),  
  print_on_error = aws_get_print_on_error(),  
  retry_time = aws_get_retry_time(),  
  network_timeout = aws_get_network_timeout(),  
  region = aws_get_region()
)
```
Arguments

TransitGatewayRouteTableId
Character. The ID of the route table.

S3Bucket
Character. The name of the S3 bucket.

Filter
Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value
A list object or a character vector

TransitGatewayRouteTableId
The ID of the route table.

S3Bucket
The name of the S3 bucket.

Filter
One or more filters. The possible values are:

• attachment.transit-gateway-attachment-id - The id of the transit gateway attachment.
• attachment.resource-id - The resource id of the transit gateway attachment.
• route-search.exact-match - The exact match of the specified filter.
• route-search.longest-prefix-match - The longest prefix that matches the route.
• route-search.subnet-of-match - The routes with a subnet that match the specified CIDR filter.
• route-search.supernet-of-match - The routes with a CIDR that encompass the CIDR filter. For example, if you have 10.0.1.0/29 and 10.0.1.0/31 routes in your route table and you specify supernet-of-match as 10.0.1.0/30, then the result returns 10.0.1.0/29.
• **state** - The state of the route (active | blackhole).
• **transit-gateway-route-destination-cidr-block** - The CIDR range.
• **type** - The type of route (propagated | static).

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**ec2_get_associated_enclave_certificate_iam_roles**

*Get Associated Enclave Certificate Iam Roles*

**Description**

Returns the IAM roles that are associated with the specified AWS Certificate Manager (ACM) certificate. It also returns the name of the Amazon S3 bucket and the Amazon S3 object key where the certificate, certificate chain, and encrypted private key bundle are stored, and the ARN of the AWS Key Management Service (KMS) customer master key (CMK) that’s used to encrypt the private key.

**Usage**

```r
ec2_get_associated_enclave_certificate_iam_roles(
  CertificateArn = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- **CertificateArn**  Character. The ARN of the ACM certificate for which to view the associated IAM roles, encryption keys, and Amazon...
- **DryRun**  Logical. Checks whether you have the required permissions for the action, without actually making the request,...
- **simplify**  Logical. Whether to simplify the result and handle nextToken in the response
- **others**  Named list. The parameters that are not included in the function parameters and need to be added into the request
- **print_on_error**  Logical. Whether to show an error message when a network error occurs.
Description

Gets information about the IPv6 CIDR block associations for a specified IPv6 address pool.

Usage

```r
ec2_get_associated_ipv6_pool_cidrs(
  PoolId,
  NextToken = NULL,
  MaxResults = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
Arguments

- **PoolId**
  Character. The ID of the IPv6 address pool.

- **NextToken**
  Characters. The token for the next page of results[optional]

- **MaxResults**
  Integer. The maximum number of results to return with a single call.[optional]

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  Character. The region of the AWS service.

Value

A list object or a character vector

**PoolId**

The ID of the IPv6 address pool.

**MaxResults**

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
ec2_get_capacity_reservation_usage

Get Capacity Reservation Usage

Description

Gets usage information about a Capacity Reservation. If the Capacity Reservation is shared, it shows usage information for the Capacity Reservation owner and each AWS account that is currently using the shared capacity. If the Capacity Reservation is not shared, it shows only the Capacity Reservation owner's usage.

Usage

```r
ec2_get_capacity_reservation_usage(
  CapacityReservationId,
  NextToken = NULL,
  MaxResults = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **CapacityReservationId**: Character. The ID of the Capacity Reservation.
- **NextToken**: Characters. The token for the next page of results.[optional]
- **MaxResults**: Integer. The maximum number of results to return for the request in a single page.[optional]
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]
- **simplify**: Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.
Value

A list object or a character vector

CapacityReservationId

The ID of the Capacity Reservation.

MaxResults

The maximum number of results to return for the request in a single page. The remaining results can be seen by sending another request with the returned nextToken value. This value can be between 5 and 500. If maxResults is given a larger value than 500, you receive an error.

Valid range: Minimum value of 1. Maximum value of 1000.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_get_coip_pool_usage

Get Coip Pool Usage

Description

Describes the allocations from the specified customer-owned address pool.

Usage

```r
ec2_get_coip_pool_usage(
  PoolId,
  Filter = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
ec2_get_coip_pool_usage

Arguments

- **PoolId**: Character. The ID of the address pool.
- **Filter**: Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below) [optional]
- **MaxResults**: Integer. The maximum number of results to return with a single call. [optional]
- **NextToken**: Characters. The token for the next page of results [optional]
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request,... [optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response [optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

**PoolId**

The ID of the address pool.

**Filter**

The filters. The following are the possible values:

- `coip-address-usage.allocation-id`
- `coip-address-usage.aws-account-id`
- `coip-address-usage.aws-service`
- `coip-address-usage.co-ip`
MaxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

dc2_get_console_output

Get Console Output

Description

Get Console Output

Usage

```
ec2_get_console_output(
    InstanceId,
    DryRun = NULL,
    Latest = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>InstanceId</td>
<td>Character. The ID of the instance.</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]</td>
</tr>
<tr>
<td>Latest</td>
<td>Logical. When enabled, retrieves the latest console output for the instance. [optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response [optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not able to get the response, an error will be thrown.</td>
</tr>
</tbody>
</table>
network_timeout

   Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region

   Character. The region of the AWS service.

Value

   A list object or a character vector

InstanceId

   The ID of the instance.

DryRun

   Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Latest

   When enabled, retrieves the latest console output for the instance.
   Default: disabled (false)

ec2_get_console_screenshot

   Get Console Screenshot

Description

   Get Console Screenshot

Usage

   ec2_get_console_screenshot(
      InstanceId,
      DryRun = NULL,
      WakeUp = NULL,
      simplify = TRUE,
      others = list(),
      print_on_error = aws_get_print_on_error(),
      retry_time = aws_get_retry_time(),
      network_timeout = aws_get_network_timeout(),
      region = aws_get_region()
   )
Arguments

InstanceId  Character. The ID of the instance.
DryRun  Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]
WakeUp  Logical. When set to true, acts as keystroke input and wakes up an instance that's in standby or 'sleep'...[optional]
simplify  Logical. Whether to simplify the result and handle nextToken in the response[optional]
others  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error  Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region  Character. The region of the AWS service.

Value

A list object or a character vector

InstanceId

The ID of the instance.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

WakeUp

When set to true, acts as keystroke input and wakes up an instance that's in standby or 'sleep' mode.
ec2_get_default_credit_specification

Get Default Credit Specification

Description

Get Default Credit Specification

Usage

```r
ec2_get_default_credit_specification(
  InstanceFamily,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **InstanceFamily**: Character. The instance family.
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response[optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

InstanceFamily

The instance family.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_get_ebs_default_kms_key_id

Get Ebs Default Kms Key Id

Description

Get Ebs Default Kms Key Id

Usage

ce2_get_ebs_default_kms_key_id(
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value

A list object or a character vector
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description

Get Ebs Encryption By Default

Usage

ec2_get_ebs_encryption_by_default(
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value

A list object or a character vector
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**ec2_get_groups_for_capacityReservation**

*Get Groups For Capacity Reservation*

**Description**

Lists the resource groups to which a Capacity Reservation has been added.

**Usage**

```r
ec2_get_groups_for_capacity_reservation(
  CapacityReservationId,
  NextToken = NULL,
  MaxResults = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- **CapacityReservationId**
  - Character. The ID of the Capacity Reservation.
- **NextToken**
  - Characters. The token for the next page of results.[optional]
- **MaxResults**
  - Integer. The maximum number of results to return for the request in a single page.[optional]
- **DryRun**
  - Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
- **simplify**
  - Logical. Whether to simplify the result and handle nextToken in the response.[optional]
- **others**
  - Named list. The parameters that are not included in the function parameters and need to be added into the request.[optional]
- **print_on_error**
  - Logical. Whether to show an error message when a network error occurs.
- **retry_time**
  - Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
Get Host Reservation Purchase Preview

**Description**

Get Host Reservation Purchase Preview

**Usage**

```r
e2_get_host_reservation_purchase_preview(
  HostIdSet,
  OfferingId,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
ec2_get_launch_template_data

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HostIdSet</td>
<td>List. The IDs of the Dedicated Hosts with which the reservation is associated.</td>
</tr>
<tr>
<td>OfferingId</td>
<td>Character. The offering ID of the reservation.</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response[optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>

Value

A list object or a character vector

HostIdSet

The IDs of the Dedicated Hosts with which the reservation is associated.

OfferingId

The offering ID of the reservation.

---

**ec2_get_launch_template_data**

*Get Launch Template Data*

---

Description

Get Launch Template Data

Usage

```r
ec2_get_launch_template_data(
  InstanceId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
Arguments

InstanceId Character. The ID of the instance.

DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request, ... [optional]

simplify Logical. Whether to simplify the result and handle nextToken in the response [optional]

others Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.

Value

A list object or a character vector

InstanceId

The ID of the instance.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description

Gets information about the resources that are associated with the specified managed prefix list.
Usage

```
ec2_get_managed_prefix_list_associations(
    PrefixListId,
    DryRun = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **PrefixListId**: Character. The ID of the prefix list.
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **MaxResults**: Integer. The maximum number of results to return with a single call. [optional]
- **NextToken**: Characters. The token for the next page of results. [optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

**PrefixListId**

The ID of the prefix list.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.
MaxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

Description

Gets information about the entries for a specified managed prefix list.

Usage

c2_get_managed_prefix_list_entries(
    PrefixListId,
    DryRun = NULL,
    TargetVersion = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

PrefixListId Character. The ID of the prefix list.
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
TargetVersion Integer. The version of the prefix list for which to return the entries. The default is the current version.[optional]
MaxResults Integer. The maximum number of results to return with a single call.[optional]
NextToken Characters. The token for the next page of results[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
Description

Get Password Data

Usage

```r
e2_get_password_data(
  InstanceId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)```
ec2_get_reserved_instances_exchange_quote

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>InstanceId</td>
<td>Character. The ID of the Windows instance.</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request, and provides an error response.</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response.</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request.</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>

Value

A list object or a character vector

InstanceId

The ID of the Windows instance.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_get_reserved_instances_exchange_quote

Get Reserved Instances Exchange Quote

Description

Returns a quote and exchange information for exchanging one or more specifiedConvertible Reserved Instances for a new Convertible Reserved Instance. If the exchange cannot be performed, the reason is returned in the response. Use AcceptReservedInstancesExchangeQuote to perform the exchange.
Usage

ec2_get_reserved_instances_exchange_quote(
    ReservedInstanceId,
    DryRun = NULL,
    TargetConfiguration = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

ReservedInstanceId
    List. The IDs of the Convertible Reserved Instances to exchange.

DryRun
    Logical. Checks whether you have the required permissions for the action, without actually
    making the request. Optional.

TargetConfiguration
    List. The configuration of the target Convertible Reserved Instance to exchange.
    for your current Convertible. Optional.

simplify
    Logical. Whether to simplify the result and handle nextToken in the response. Optional.

others
    Named list. The parameters that are not included in the function parameters and need to be
    added into the request. Optional.

print_on_error
    Logical. Whether to show an error message when a network error occurs.

retry_time
    Integer. Number of retries for a REST request when encounter the network issue. If the
    request has been sent retry_time times but still not be able to get the response,
    an error will be thrown.

network_timeout
    Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
    Character. The region of the AWS service.

Value

A list object or a character vector

ReservedInstanceId
    The IDs of the Convertible Reserved Instances to exchange.

DryRun
    Checks whether you have the required permissions for the action, without actually making the
    request, and provides an error response. If you have the required permissions, the error response
    is DryRunOperation. Otherwise, it is UnauthorizedOperation.
**TargetConfiguration**

The configuration of the target Convertible Reserved Instance to exchange for your current Convertible Reserved Instances.

---

**Description**

Lists the route tables to which the specified resource attachment propagates routes.

**Usage**

```r
ec2_get_transit_gateway_attachment_propagations(
  TransitGatewayAttachmentId,
  Filter = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- **TransitGatewayAttachmentId**
  Character. The ID of the attachment.

- **Filter**
  Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]

- **MaxResults**
  Integer. The maximum number of results to return with a single call.[optional]

- **NextToken**
  Characters. The token for the next page of results[optional]

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout
   Numeric. Number of seconds to wait for a REST response until giving up. Can
   not be less than 1 ms.

region
   Character. The region of the AWS service.

Value
   A list object or a character vector

TransitGatewayAttachmentId
   The ID of the attachment.

Filter
   One or more filters. The possible values are:

   • transit-gateway-route-table-id - The ID of the transit gateway route table.

MaxResults
   The maximum number of results to return with a single call. To retrieve the remaining results, make
   another call with the returned nextToken value.

DryRun
   Checks whether you have the required permissions for the action, without actually making the
   request, and provides an error response. If you have the required permissions, the error response is
   DryRunOperation. Otherwise, it is UnauthorizedOperation.
Arguments

TransitGatewayMulticastDomainId
   Character. The ID of the transit gateway multicast domain.[optional]

Filter
   Named list where the name is the filter name and the value is the filter's value.
   The value can be a vector or a list object (see below)[optional]

MaxResults
   Integer. The maximum number of results to return with a single call.[optional]

NextToken
   Characters. The token for the next page of results[optional]

DryRun
   Logical. Checks whether you have the required permissions for the action, without actually making the request,[optional]

simplify
   Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
   Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
   Logical. Whether to show an error message when a network error occurs.

retry_time
   Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
   Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
   Character. The region of the AWS service.

Value

A list object or a character vector

TransitGatewayMulticastDomainId

The ID of the transit gateway multicast domain.

Filter

One or more filters. The possible values are:

- resource-id - The ID of the resource.
- resource-type - The type of resource. The valid value is: vpc.
- state - The state of the subnet association. Valid values are associated | associating | disassociated | disassociating.
- subnet-id - The ID of the subnet.
- transit-gateway-attachment-id - The id of the transit gateway attachment.
Description

Gets information about the prefix list references in a specified transit gateway route table.

Usage

```r
ec2_get_transit_gateway_prefix_list_references(
  TransitGatewayRouteTableId,
  Filter = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **TransitGatewayRouteTableId**
  Character. The ID of the transit gateway route table.

- **Filter**
  Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below).[optional]

- **MaxResults**
  Integer. The maximum number of results to return with a single call.[optional]

- **NextToken**
  Characters. The token for the next page of results[optional]

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response[optional]
ec2_get_transit_gateway_prefix_list_references

others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.

Value

A list object or a character vector

TransitGatewayRouteTableId

The ID of the transit gateway route table.

Filter

One or more filters. The possible values are:

• attachment.resource-id - The ID of the resource for the attachment.
• attachment.resource-type - The type of resource for the attachment. Valid values are vpc \ vpn \ direct-connect-gateway \ peering.
• attachment.transit-gateway-attachment-id - The ID of the attachment.
• is-blackhole - Whether traffic matching the route is blocked (true \ false).
• prefix-list-id - The ID of the prefix list.
• prefix-list-owner-id - The ID of the owner of the prefix list.
• state - The state of the prefix list reference (pending \ available \ modifying \ deleting).

MaxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
Description

Gets information about the associations for the specified transit gateway route table.

Usage

```r
ec2_get_transit_gateway_route_table_associations(
  TransitGatewayRouteTableId,
  Filter = NULL,
  MaxResults = NULL,
  NextToken = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **TransitGatewayRouteTableId**: Character. The ID of the transit gateway route table.
- **Filter**: Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]
- **MaxResults**: Integer. The maximum number of results to return with a single call.[optional]
- **NextToken**: Characters. The token for the next page of results[optional]
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]
- **simplify**: Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.
- **region**: Character. The region of the AWS service.
ec2_get_transit_gateway_route_table_propagations

Value

A list object or a character vector

TransitGatewayRouteTableId

The ID of the transit gateway route table.

Filter

One or more filters. The possible values are:

• resource-id - The ID of the resource.
• resource-type - The resource type. Valid values are vpc \ vpn \ direct-connect-gateway \ peering \ connect.
• transit-gateway-attachment-id - The ID of the attachment.

MaxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Usage

```r
ec2_get_transit_gateway_route_table_propagations(
    TransitGatewayRouteTableId,
    Filter = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
)```
Arguments

TransitGatewayRouteTableId
Character. The ID of the transit gateway route table.

Filter
Named list where the name is the filter name and the value is the filter’s value.
The value can be a vector or a list object (see below)[optional]

MaxResults
Integer. The maximum number of results to return with a single call.[optional]

NextToken
Characters. The token for the next page of results[optional]

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value

A list object or a character vector

TransitGatewayRouteTableId
The ID of the transit gateway route table.

Filter
One or more filters. The possible values are:

• resource-id - The ID of the resource.
• resource-type - The resource type. Valid values are vpc ∨ vpn ∨ direct-connect-gateway ∨ peering ∨ connect.
• transit-gateway-attachment-id - The ID of the attachment.

MaxResults
The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description

Import Client Vpn Client Certificate Revocation List

Usage

```r
ec2_import_client_vpn_client_certificate_revocation_list(
  ClientVpnEndpointId,
  CertificateRevocationList,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **ClientVpnEndpointId**: Character. The ID of the Client VPN endpoint to which the client certificate revocation list applies.
- **CertificateRevocationList**: Character. The client certificate revocation list file.
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]
- **simplify**: Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.
Value
A list object or a character vector

ClientVpnEndpointId
The ID of the Client VPN endpoint to which the client certificate revocation list applies.

CertificateRevocationList
The client certificate revocation list file. For more information, see Generate a Client Certificate Revocation List in the AWS Client VPN Administrator Guide.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

desc
ec2_import_image  Import Image

Description
Import single or multi-volume disk images or EBS snapshots into an Amazon Machine Image (AMI). For more information, see Importing a VM as an Image Using VM Import/Export in the VM Import/Export User Guide.

Usage
ec2_import_image(
  Architecture = NULL,
  ClientData = NULL,
  ClientToken = NULL,
  Description = NULL,
  DiskContainer = NULL,
  DryRun = NULL,
  Encrypted = NULL,
  Hypervisor = NULL,
  KmsKeyId = NULL,
  LicenseType = NULL,
  Platform = NULL,
  RoleName = NULL,
  LicenseSpecifications = NULL,
  TagSpecification = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
)
```python
class ec2_import_image:
    def __init__(self):
        self.retry_time = aws_get_retry_time(),
        self.network_timeout = aws_get_network_timeout(),
        self.region = aws_get_region()

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Architecture</td>
<td>Character. The architecture of the virtual machine. Valid values: i386 \ x86_64 \ arm64 [optional]</td>
</tr>
<tr>
<td>ClientData</td>
<td>Object. The client-specific data.[optional]</td>
</tr>
<tr>
<td>ClientToken</td>
<td>Character. The token to enable idempotency for VM import requests.[optional]</td>
</tr>
<tr>
<td>Description</td>
<td>Character. A description string for the import image task.[optional]</td>
</tr>
<tr>
<td>DiskContainer</td>
<td>List. Information about the disk containers.[optional]</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]</td>
</tr>
<tr>
<td>Encrypted</td>
<td>Logical. Specifies whether the destination AMI of the imported image should be encrypted.[optional]</td>
</tr>
<tr>
<td>Hypervisor</td>
<td>Character. The target hypervisor platform. Valid values: xen [optional]</td>
</tr>
<tr>
<td>KmsKeyId</td>
<td>Character. An identifier for the symmetric AWS Key Management Service (AWS KMS) customer master key (CMK)...[optional]</td>
</tr>
<tr>
<td>LicenseType</td>
<td>Character. The license type to be used for the Amazon Machine Image (AMI) after importing.[optional]</td>
</tr>
<tr>
<td>Platform</td>
<td>Character. The operating system of the virtual machine. Valid values: windows \ Linux [optional]</td>
</tr>
<tr>
<td>RoleName</td>
<td>Character. The name of the role to use when not using the default role, \vmimport.[optional]</td>
</tr>
<tr>
<td>LicenseSpecifications</td>
<td>List. The ARNs of the license configurations.[optional]</td>
</tr>
<tr>
<td>TagSpecification</td>
<td>List. The tags to apply to the import image task during creation.[optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response[optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>

Value

A list object or a character vector
Architecture

The architecture of the virtual machine.
Valid values: i386 | x86_64 | arm64

ClientData

The client-specific data.

ClientToken

The token to enable idempotency for VM import requests.

Description

A description string for the import image task.

DiskContainer

Information about the disk containers.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Encrypted

Specifies whether the destination AMI of the imported image should be encrypted. The default CMK for EBS is used unless you specify a non-default AWS Key Management Service (AWS KMS) CMK using KmsKeyId. For more information, see Amazon EBS Encryption in the Amazon Elastic Compute Cloud User Guide.

Hypervisor

The target hypervisor platform.
Valid values: xen

KmsKeyId

An identifier for the symmetric AWS Key Management Service (AWS KMS) customer master key (CMK) to use when creating the encrypted AMI. This parameter is only required if you want to use a non-default CMK; if this parameter is not specified, the default CMK for EBS is used. If a KmsKeyId is specified, the Encrypted flag must also be set.
The CMK identifier may be provided in any of the following formats:

- Key ID
Key alias. The alias ARN contains the `arn:aws:kms` namespace, followed by the Region of the CMK, the AWS account ID of the CMK owner, the alias namespace, and then the CMK alias. For example, `arn:aws:kms:us-east-1:012345678910:alias/ExampleAlias`.

ARN using key ID. The ID ARN contains the `arn:aws:kms` namespace, followed by the Region of the CMK, the AWS account ID of the CMK owner, the key namespace, and then the CMK ID. For example, `arn:aws:kms:us-east-1:012345678910:key/abcd1234-a123-456a-a12b-a123b4cd56ef`.

ARN using key alias. The alias ARN contains the `arn:aws:kms` namespace, followed by the Region of the CMK, the AWS account ID of the CMK owner, the alias namespace, and then the CMK alias. For example, `arn:aws:kms:us-east-1:012345678910:alias/ExampleAlias`.

AWS parses `kmsKeyId` asynchronously, meaning that the action you call may appear to complete even though you provided an invalid identifier. This action will eventually report failure.

The specified CMK must exist in the Region that the AMI is being copied to.

Amazon EBS does not support asymmetric CMKs.

**LicenseType**

The license type to be used for the Amazon Machine Image (AMI) after importing.

By default, we detect the source-system operating system (OS) and apply the appropriate license. Specify `AWS` to replace the source-system license with an AWS license, if appropriate. Specify `BYOL` to retain the source-system license, if appropriate.

To use `BYOL`, you must have existing licenses with rights to use these licenses in a third party cloud, such as AWS. For more information, see Prerequisites in the VM Import/Export User Guide.

**Platform**

The operating system of the virtual machine.

Valid values: `Windows` | `Linux`

**RoleName**

The name of the role to use when not using the default role, `vmimport`.

**LicenseSpecifications**

The ARNs of the license configurations.

**TagSpecification**

The tags to apply to the import image task during creation.
ec2_import_instance  Import Instance

Description

Import Instance

Usage

e2_import_instance(
    Platform,
    Description = NULL,
    DiskImage = NULL,
    DryRun = NULL,
    LaunchSpecification = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

Platform  Character. The instance operating system.
Description  Character. A description for the instance being imported.[optional]
DiskImage  List. The disk image.[optional]
DryRun  Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]
LaunchSpecification  Object. The launch specification.[optional]
simplify  Logical. Whether to simplify the result and handle nextToken in the response[optional]
others  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error  Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region  Character. The region of the AWS service.
Value
A list object or a character vector

Platform
The instance operating system.

Description
A description for the instance being imported.

DiskImage
The disk image.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

LaunchSpecification
The launch specification.

---

**ec2_import_key_pair**

*Import Key Pair*

**Description**
Import Key Pair

**Usage**

```r
ec2_import_key_pair(
  KeyName,
  PublicKeyMaterial,
  DryRun = NULL,
  TagSpecification = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
**Arguments**

KeyName: Character. A unique name for the key pair.

PublicKeyMaterial: Character. The public key.

DryRun: Logical. Checks whether you have the required permissions for the action, without actually making the request. (optional)

TagSpecification: List. The tags to apply to the imported key pair. (optional)

simplify: Logical. Whether to simplify the result and handle nextToken in the response. (optional)

others: Named list. The parameters that are not included in the function parameters and need to be added into the request. (optional)

print_on_error: Logical. Whether to show an error message when a network error occurs.

retry_time: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region: Character. The region of the AWS service.

**Value**

A list object or a character vector

**KeyName**

A unique name for the key pair.

**PublicKeyMaterial**

The public key. For API calls, the text must be base64-encoded. For command line tools, base64 encoding is performed for you.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

**TagSpecification**

The tags to apply to the imported key pair.
ec2_import_snapshot  Import Snapshot

Description
Imports a disk into an EBS snapshot.

Usage
ec2_import_snapshot(
    ClientData = NULL,
    ClientToken = NULL,
    Description = NULL,
    DiskContainer = NULL,
    DryRun = NULL,
    Encrypted = NULL,
    KmsKeyId = NULL,
    RoleName = NULL,
    TagSpecification = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

ClientData  Object. The client-specific data.[optional]
ClientToken  Character. Token to enable idempotency for VM import requests.[optional]
Description  Character. The description string for the import snapshot task.[optional]
DiskContainer  Object. Information about the disk container.[optional]
DryRun  Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
Encrypted  Logical. Specifies whether the destination snapshot of the imported image should be encrypted.[optional]
KmsKeyId  Character. An identifier for the symmetric AWS Key Management Service (AWS KMS) customer master key (CMK)...[optional]
RoleName  Character. The name of the role to use when not using the default role, \'vmimport\'.[optional]
TagSpecification  List. The tags to apply to the import snapshot task during creation.[optional]
simplify  Logical. Whether to simplify the result and handle nextToken in the response[optional]
ec2_import_snapshot

others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.

Value

A list object or a character vector

ClientData

The client-specific data.

ClientToken

Token to enable idempotency for VM import requests.

Description

The description string for the import snapshot task.

DiskContainer

Information about the disk container.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Encrypted

Specifies whether the destination snapshot of the imported image should be encrypted. The default CMK for EBS is used unless you specify a non-default AWS Key Management Service (AWS KMS) CMK using KmsKeyId. For more information, see Amazon EBS Encryption in the Amazon Elastic Compute Cloud User Guide.
KmsKeyId

An identifier for the symmetric AWS Key Management Service (AWS KMS) customer master key (CMK) to use when creating the encrypted snapshot. This parameter is only required if you want to use a non-default CMK; if this parameter is not specified, the default CMK for EBS is used. If a KmsKeyId is specified, the Encrypted flag must also be set.

The CMK identifier may be provided in any of the following formats:

- Key ID
- Key alias. The alias ARN contains the arn:aws:kms namespace, followed by the Region of the CMK, the AWS account ID of the CMK owner, the alias namespace, and then the CMK alias. For example, arn:aws:kms:us-east-1:012345678910:alias/ExampleAlias.
- ARN using key ID. The ID ARN contains the arn:aws:kms namespace, followed by the Region of the CMK, the AWS account ID of the CMK owner, the key namespace, and then the CMK ID. For example, arn:aws:kms:us-east-1:012345678910:key/abcd1234-a123-456a-a12b-a123b4cd56ef.
- ARN using key alias. The alias ARN contains the arn:aws:kms namespace, followed by the Region of the CMK, the AWS account ID of the CMK owner, the alias namespace, and then the CMK alias. For example, arn:aws:kms:us-east-1:012345678910:alias/ExampleAlias.

AWS parses KmsKeyId asynchronously, meaning that the action you call may appear to complete even though you provided an invalid identifier. This action will eventually report failure.

The specified CMK must exist in the Region that the snapshot is being copied to.

Amazon EBS does not support asymmetric CMKs.

RoleName

The name of the role to use when not using the default role, `vmimport`.

TagSpecification

The tags to apply to the import snapshot task during creation.
Usage

```r
e2_import_volume(
  AvailabilityZone,
  Image,
  Volume,
  Description = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **AvailabilityZone** Character. The Availability Zone for the resulting EBS volume.
- **Image** Object. The disk image.
- **Volume** Object. The volume size.
- **Description** Character. A description of the volume.[optional]
- **DryRun** Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
- **simplify** Logical. Whether to simplify the result and handle `nextToken` in the response[optional]
- **others** Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error** Logical. Whether to show an error message when a network error occurs.
- **retry_time** Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout** Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region** Character. The region of the AWS service.

Value

A list object or a character vector

**AvailabilityZone**

The Availability Zone for the resulting EBS volume.

**Image**

The disk image.
**Volume**

The volume size.

**Description**

A description of the volume.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**ec2_modify_address_attribute**

*Modify Address Attribute*

---

**Description**

Modifies an attribute of the specified Elastic IP address. For requirements, see Using reverse DNS for email applications.

**Usage**

```r
ec2_modify_address_attribute(
  AllocationId,
  DomainName = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AllocationId</td>
<td>Character. [EC2-VPC] The allocation ID.</td>
</tr>
<tr>
<td>DomainName</td>
<td>Character. The domain name to modify for the IP address.[optional]</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response[optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]</td>
</tr>
</tbody>
</table>
print_on_error       Logical. Whether to show an error message when a network error occurs.
retry_time          Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout     Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region              Character. The region of the AWS service.

Value
A list object or a character vector

AllocationId
[EC2-VPC] The allocation ID.

DomainName
The domain name to modify for the IP address.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_modify_availability_zone_group
Modify Availability Zone Group

Description
Modify Availability Zone Group

Usage
ec2_modify_availability_zone_group(
  GroupName,
  OptInStatus,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
Arguments

GroupBox  Character. The name of the Availability Zone group, Local Zone group, or Wavelength Zone group.

OptInStatus  Character. Indicates whether you are opted in to the Local Zone group or Wavelength Zone group.

DryRun  Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]

simplify  Logical. Whether to simplify the result and handle nextToken in the response[optional]

others  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error  Logical. Whether to show an error message when a network error occurs.

retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  Character. The region of the AWS service.

Value

A list object or a character vector

GroupBox  The name of the Availability Zone group, Local Zone group, or Wavelength Zone group.

OptInStatus  Indicates whether you are opted in to the Local Zone group or Wavelength Zone group. The only valid value is opted-in. You must contact AWS Support to opt out of a Local Zone group, or Wavelength Zone group.

DryRun  Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
Description

Modifies a Capacity Reservation’s capacity and the conditions under which it is to be released. You cannot change a Capacity Reservation’s instance type, EBS optimization, instance store settings, platform, Availability Zone, or instance eligibility. If you need to modify any of these attributes, we recommend that you cancel the Capacity Reservation, and then create a new one with the required attributes.

Usage

```
ec2_modify_capacity_reservation(
    CapacityReservationId,  
    InstanceCount = NULL,  
    EndDate = NULL,  
    EndDateType = NULL,  
    Accept = NULL,  
    DryRun = NULL,  
    simplify = TRUE,  
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **CapacityReservationId**
  Character. The ID of the Capacity Reservation.
- **InstanceCount**
  Integer. The number of instances for which to reserve capacity.[optional]
- **EndDate**
  Character. The date and time at which the Capacity Reservation expires.[optional]
- **EndDateType**
  Character. Indicates the way in which the Capacity Reservation ends.[optional]
- **Accept**
  Logical. Reserved. Capacity Reservations you have created are accepted by default.[optional]
- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request,[optional]
- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  Character. The region of the AWS service.

Value
A list object or a character vector

CapacityReservationId
The ID of the Capacity Reservation.

InstanceCount
The number of instances for which to reserve capacity.

EndDate
The date and time at which the Capacity Reservation expires. When a Capacity Reservation expires, the reserved capacity is released and you can no longer launch instances into it. The Capacity Reservation’s state changes to expired when it reaches its end date and time.

The Capacity Reservation is cancelled within an hour from the specified time. For example, if you specify 5/31/2019, 13:30:55, the Capacity Reservation is guaranteed to end between 13:30:55 and 14:30:55 on 5/31/2019.

You must provide an EndDate value if EndDateType is limited. Omit EndDate if EndDateType is unlimited.

EndDateType
Indicates the way in which the Capacity Reservation ends. A Capacity Reservation can have one of the following end types:

- unlimited - The Capacity Reservation remains active until you explicitly cancel it. Do not provide an EndDate value if EndDateType is unlimited.
- limited - The Capacity Reservation expires automatically at a specified date and time. You must provide an EndDate value if EndDateType is limited.

Accept
Reserved. Capacity Reservations you have created are accepted by default.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Usage

e2_modify_client_vpn_endpoint(ClientVpnEndpointId, ServerCertificateArn = NULL, ConnectionLogOptions = NULL, DnsServers = NULL, VpnPort = NULL, Description = NULL, SplitTunnel = NULL, DryRun = NULL, SecurityGroupId = NULL, VpcId = NULL, SelfServicePortal = NULL, ClientConnectOptions = NULL, simplify = TRUE, others = list(), print_on_error = aws_get_print_on_error(), retry_time = aws_get_retry_time(), network_timeout = aws_get_network_timeout(), region = aws_get_region())

Arguments

ClientVpnEndpointId
   Character. The ID of the Client VPN endpoint to modify.

ServerCertificateArn
   Character. The ARN of the server certificate to be used.[optional]

ConnectionLogOptions
   Object. Information about the client connection logging options.[optional]
DnsServers: Object. Information about the DNS servers to be used by Client VPN connections. [optional]

VpnPort: Integer. The port number to assign to the Client VPN endpoint for TCP and UDP traffic. [optional]

Description: Character. A brief description of the Client VPN endpoint. [optional]

SplitTunnel: Logical. Indicates whether the VPN is split-tunnel. [optional]

DryRun: Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]

SecurityGroupId: List. The IDs of one or more security groups to apply to the target network. [optional]

VpcId: Character. The ID of the VPC to associate with the Client VPN endpoint. [optional]

SelfServicePortal: Character. Specify whether to enable the self-service portal for the Client VPN endpoint. [optional]

ClientConnectOptions: Object. The options for managing connection authorization for new client connections. [optional]

simplify: Logical. Whether to simplify the result and handle nextToken in the response. [optional]

others: Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]

print_on_error: Logical. Whether to show an error message when a network error occurs.

retry_time: Integer. Number of retries for a REST request when encountering the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout: Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.

region: Character. The region of the AWS service.

Value

A list object or a character vector

ClientVpnEndpointId

The ID of the Client VPN endpoint to modify.

ServerCertificateArn

The ARN of the server certificate to be used. The server certificate must be provisioned in AWS Certificate Manager (ACM).
ConnectionLogOptions

Information about the client connection logging options. If you enable client connection logging, data about client connections is sent to a Cloudwatch Logs log stream. The following information is logged:

- Client connection requests
- Client connection results (successful and unsuccessful)
- Reasons for unsuccessful client connection requests
- Client connection termination time

DnsServers

Information about the DNS servers to be used by Client VPN connections. A Client VPN endpoint can have up to two DNS servers.

VpnPort

The port number to assign to the Client VPN endpoint for TCP and UDP traffic.

Valid Values: 443 | 1194
Default Value: 443

Description

A brief description of the Client VPN endpoint.

SplitTunnel

Indicates whether the VPN is split-tunnel.

For information about split-tunnel VPN endpoints, see Split-Tunnel AWS Client VPN Endpoint in the AWS Client VPN Administrator Guide.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

SecurityGroupId

The IDs of one or more security groups to apply to the target network.

VpcId

The ID of the VPC to associate with the Client VPN endpoint.

SelfServicePortal

Specify whether to enable the self-service portal for the Client VPN endpoint.
ClientConnectOptions

The options for managing connection authorization for new client connections.

ec2_modify_default_credit_specification

Modify Default Credit Specification

Description

Modify Default Credit Specification

Usage

ec2_modify_default_credit_specification(
    InstanceFamily,
    CpuCredits,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>InstanceFamily</td>
<td>Character. The instance family.</td>
</tr>
<tr>
<td>CpuCredits</td>
<td>Character. The credit option for CPU usage of the instance family.</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response[optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>
**Value**

A list object or a character vector

**InstanceFamily**

The instance family.

**CpuCredits**

The credit option for CPU usage of the instance family.

Valid Values: `standard` \| `unlimited`

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

---

**ec2_modify_ebs_default_kms_key_id**

*Modify Ebs Default Kms Key Id*

---

**Description**

Modify Ebs Default Kms Key Id

**Usage**

```r
ec2_modify_ebs_default_kms_key_id(
    KmsKeyId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

**Arguments**

- **KmsKeyId**
  Character. The identifier of the AWS Key Management Service (AWS KMS) customer master key (CMK) to use for...

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response[optional]
ec2_modify_ebs_default_kms_key_id

others
   Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error
   Logical. Whether to show an error message when a network error occurs.
retry_time
   Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout
   Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region
   Character. The region of the AWS service.

Value

A list object or a character vector

KmsKeyId

The identifier of the AWS Key Management Service (AWS KMS) customer master key (CMK) to use for Amazon EBS encryption. If this parameter is not specified, your AWS managed CMK for EBS is used. If KmsKeyId is specified, the encrypted state must be true.

You can specify the CMK using any of the following:

- Key ID. For example, 1234abcd-12ab-34cd-56ef-1234567890ab.
- Key alias. For example, alias/ExampleAlias.
- Key ARN. For example, arn:aws:kms:us-east-1:012345678910:key/1234abcd-12ab-34cd-56ef-1234567890ab.
- Alias ARN. For example, arn:aws:kms:us-east-1:012345678910:alias/ExampleAlias.

AWS authenticates the CMK asynchronously. Therefore, if you specify an ID, alias, or ARN that is not valid, the action can appear to complete, but eventually fails.

Amazon EBS does not support asymmetric CMKs.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
Description

Modify Fleet

Usage

```r
ec2_modify_fleet(
  FleetId,
  DryRun = NULL,
  ExcessCapacityTerminationPolicy = NULL,
  LaunchTemplateConfig = NULL,
  TargetCapacitySpecification = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **FleetId**: Character. The ID of the EC2 Fleet.
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **ExcessCapacityTerminationPolicy**: Character. Indicates whether running instances should be terminated if the total target capacity of the EC2... [optional]
- **LaunchTemplateConfig**: List. The launch template and overrides. [optional]
- **TargetCapacitySpecification**: Object. The size of the EC2 Fleet. [optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response [optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.
- **region**: Character. The region of the AWS service.
Value

A list object or a character vector

FleetId

The ID of the EC2 Fleet.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ExcessCapacityTerminationPolicy

Indicates whether running instances should be terminated if the total target capacity of the EC2 Fleet is decreased below the current size of the EC2 Fleet.

LaunchTemplateConfig

The launch template and overrides.

TargetCapacitySpecification

The size of the EC2 Fleet.

---

**ec2_modify_fpga_image_attribute**

*Modify Fpga Image Attribute*

---

**Description**

Modifies the specified attribute of the specified Amazon FPGA Image (AFI).

**Usage**

```r
ec2_modify_fpga_image_attribute(  
  FpgaImageId,  
  DryRun = NULL,  
  Attribute = NULL,  
  OperationType = NULL,  
  UserId = NULL,  
  UserGroup = NULL,  
  ProductCode = NULL,  
  LoadPermission = NULL,  
  Description = NULL,  
  Name = NULL,  
)```
simplify = TRUE,
others = list(),
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

FpgaImageId: Character. The ID of the AFI.
DryRun: Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
Attribute: Character. The name of the attribute.[optional]
OperationType: Character. The operation type.[optional]
UserId: List. The AWS account IDs. This parameter is valid only when modifying the loadPermission attribute.[optional]
UserGroup: List. The user groups. This parameter is valid only when modifying the loadPermission attribute.[optional]
ProductCode: List. The product codes.[optional]
LoadPermission: Object. The load permission for the AFI.[optional]
Description: Character. A description for the AFI.[optional]
Name: Character. A name for the AFI.[optional]
simplify: Logical. Whether to simplify the result and handle nextToken in the response.[optional]
others: Named list. The parameters that are not included in the function parameters and need to be added into the request.[optional]
print_on_error: Logical. Whether to show an error message when a network error occurs.
retry_time: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region: Character. The region of the AWS service.

Value

A list object or a character vector

FpgaImageId

The ID of the AFI.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Attribute

The name of the attribute.

OperationType

The operation type.

UserId

The AWS account IDs. This parameter is valid only when modifying the loadPermission attribute.

UserGroup

The user groups. This parameter is valid only when modifying the loadPermission attribute.

ProductCode

The product codes. After you add a product code to an AFI, it can't be removed. This parameter is valid only when modifying the productCodes attribute.

LoadPermission

The load permission for the AFI.

Description

A description for the AFI.

Name

A name for the AFI.
**ec2_modify_hosts**  
Modify Hosts

**Description**
Modify Hosts

**Usage**

```r
ec2_modify_hosts(
  HostId,
  AutoPlacement = NULL,
  HostRecovery = NULL,
  InstanceType = NULL,
  InstanceFamily = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HostId</td>
<td>List. The IDs of the Dedicated Hosts to modify.</td>
</tr>
<tr>
<td>AutoPlacement</td>
<td>Character. Specify whether to enable or disable auto-placement.[optional]</td>
</tr>
<tr>
<td>HostRecovery</td>
<td>Character. Indicates whether to enable or disable host recovery for the Dedicated Host.[optional]</td>
</tr>
<tr>
<td>InstanceType</td>
<td>Character. Specifies the instance type to be supported by the Dedicated Host.[optional]</td>
</tr>
<tr>
<td>InstanceFamily</td>
<td>Character. Specifies the instance family to be supported by the Dedicated Host.[optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response[optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>
Value

A list object or a character vector

HostId

The IDs of the Dedicated Hosts to modify.

AutoPlacement

Specify whether to enable or disable auto-placement.

HostRecovery

Indicates whether to enable or disable host recovery for the Dedicated Host. For more information, see Host recovery in the Amazon EC2 User Guide.

InstanceType

Specifies the instance type to be supported by the Dedicated Host. Specify this parameter to modify a Dedicated Host to support only a specific instance type.

If you want to modify a Dedicated Host to support multiple instance types in its current instance family, omit this parameter and specify InstanceFamily instead. You cannot specify InstanceType and InstanceFamily in the same request.

InstanceFamily

Specifies the instance family to be supported by the Dedicated Host. Specify this parameter to modify a Dedicated Host to support multiple instance types within its current instance family.

If you want to modify a Dedicated Host to support a specific instance type only, omit this parameter and specify InstanceType instead. You cannot specify InstanceFamily and InstanceType in the same request.

Description

Modify Identity Id Format
Usage

ec2_modify_identity_id_format(
  PrincipalArn,
  Resource,
  UseLongIds,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

PrincipalArn  Character. The ARN of the principal, which can be an IAM user, IAM role, or the root user.
Resource       Character. The type of resource: bundle \ conversion-task \ customer-gateway \ dhcp-options \...
UseLongIds     Logical. Indicates whether the resource should use longer IDs (17-character IDs)
simplify       Logical. Whether to simplify the result and handle nextToken in the response[optional]
others         Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time     Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region         Character. The region of the AWS service.

Value

A list object or a character vector

PrincipalArn

The ARN of the principal, which can be an IAM user, IAM role, or the root user. Specify all to modify the ID format for all IAM users, IAM roles, and the root user of the account.

Resource

The type of resource: bundle \ conversion-task \ customer-gateway \ dhcp-options \ elastic-ip-allocation \ elastic-ip-association \ export-task \ flow-log \ image \ import-task \ internet-gateway
ec2_modify_id_format

network-acl \ network-acl-association \ network-interface \ network-interface-attachment
prefix-list \ route-table \ route-table-association \ security-group \ subnet
subnet-cidr-block-association \ vpc \ vpc-cidr-block-association \ vpc-endpoint
vpc-peering-connection \ vpn-connection \ vpn-gateway.
Alternatively, use the all-current option to include all resource types that are currently within their opt-in period for longer IDs.

UseLongIds
Indicates whether the resource should use longer IDs (17-character IDs)

ec2_modify_id_format (Modify Id Format)

Description
Modify Id Format

Usage
ec2_modify_id_format(
  Resource,
  UseLongIds,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments
| Resource           | Character. The type of resource: bundle \ conversion-task \ customer-gateway \ dhcp-options \...
| UseLongIds        | Logical. Indicate whether the resource should use longer IDs (17-character IDs).
| simplify          | Logical. Whether to simplify the result and handle nextToken in the response[optional]
| others            | Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
| print_on_error    | Logical. Whether to show an error message when a network error occurs.
| retry_time        | Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
| network_timeout   | Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
| region            | Character. The region of the AWS service.
Value
A list object or a character vector

Resource
The type of resource: bundle \ conversion-task \ customer-gateway \ dhcp-options \ elastic-ip-allocation
\ elastic-ip-association \ export-task \ flow-log \ image \ import-task \ internet-gateway
\ network-acl \ network-acl-association \ network-interface \ network-interface-attachment
\ prefix-list \ route-table \ route-table-association \ security-group \ subnet \ subnet-cidr-block-association \ vpc \ vpc-cidr-block-association \ vpc-endpoint \ vpc-peer-connection \ vpn-connection \ vpn-gateway.
Alternatively, use the all-current option to include all resource types that are currently within their opt-in period for longer IDs.

UseLongIds
Indicate whether the resource should use longer IDs (17-character IDs).

ec2_modify_image_attribute
Modify Image Attribute

Description
Modify Image Attribute

Usage
ec2_modify_image_attribute(
    ImageId,
    Attribute = NULL,
    Description = NULL,
    LaunchPermission = NULL,
    OperationType = NULL,
    ProductCode = NULL,
    UserGroup = NULL,
    UserId = NULL,
    Value = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
### Arguments

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ImageId</td>
<td>Character. The ID of the AMI.</td>
</tr>
<tr>
<td>Attribute</td>
<td>Character. The name of the attribute to modify.[optional]</td>
</tr>
<tr>
<td>Description</td>
<td>Object. A new description for the AMI.[optional]</td>
</tr>
<tr>
<td>LaunchPermission</td>
<td>Object. A new launch permission for the AMI.[optional]</td>
</tr>
<tr>
<td>OperationType</td>
<td>Character. The operation type.[optional]</td>
</tr>
<tr>
<td>ProductCode</td>
<td>List. The DevPay product codes. After you add a product code to an AMI, it can't be removed.[optional]</td>
</tr>
<tr>
<td>UserGroup</td>
<td>List. The user groups.[optional]</td>
</tr>
<tr>
<td>UserId</td>
<td>List. The AWS account IDs.[optional]</td>
</tr>
<tr>
<td>Value</td>
<td>Character. The value of the attribute being modified.[optional]</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response[optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>

### Value

A list object or a character vector

### ImageId

The ID of the AMI.

### Attribute

The name of the attribute to modify. The valid values are description, launchPermission, and productCodes.

### Description

A new description for the AMI.
LaunchPermission

A new launch permission for the AMI.

OperationType

The operation type. This parameter can be used only when the Attribute parameter is launchPermission.

ProductCode

The DevPay product codes. After you add a product code to an AMI, it can't be removed.

UserGroup

The user groups. This parameter can be used only when the Attribute parameter is launchPermission.

UserId

The AWS account IDs. This parameter can be used only when the Attribute parameter is launchPermission.

Value

The value of the attribute being modified. This parameter can be used only when the Attribute parameter is description or productCodes.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**ec2_modify_instance_attribute**

*Modify Instance Attribute*

---

**Description**

Modify Instance Attribute

**Usage**

```python
ec2_modify_instance_attribute(
    InstanceId,
    SourceDestCheck = NULL,
    Attribute = NULL,
    BlockDeviceMapping = NULL,
    DisableApiTermination = NULL,
    DryRun = NULL,
    EbsOptimized = NULL,
```
EnaSupport = NULL,
GroupId = NULL,
InstanceInitiatedShutdownBehavior = NULL,
InstanceType = NULL,
Kernel = NULL,
Ramdisk = NULL,
SriovNetSupport = NULL,
UserData = NULL,
Value = NULL,
simplify = TRUE,
others = list(),
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

InstanceId  Character. The ID of the instance.
SourceDestCheck  Object. Specifies whether source/destination checking is enabled.[optional]
Attribute  Character. The name of the attribute.[optional]
BlockDeviceMapping  List. Modifies the DeleteOnTermination attribute for volumes that are currently attached.[optional]
DisableApiTermination  Object. If the value is true, you can't terminate the instance using the Amazon EC2 console, CLI, or API;...[optional]
DryRun  Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
EbsOptimized  Object. Specifies whether the instance is optimized for Amazon EBS I/O.[optional]
EnaSupport  Object. Specifies whether the instance is enabled for enhanced networking with ENA for the instance.[optional]
GroupId List. [EC2-VPC] Changes the security groups of the instance.[optional]
InstanceInitiatedShutdownBehavior  Object. Specifies whether the instance stops or terminates when you initiate shutdown from the instance (using the operating system command for system shutdown)....[optional]
InstanceType  Object. Changes the instance type to the specified value.[optional]
Kernel  Object. Changes the instance's kernel to the specified value.[optional]
Ramdisk  Object. Changes the instance's RAM disk to the specified value.[optional]
SriovNetSupport  Object. Set to simple to enable enhanced networking with the Intel 82599 Virtual Function interface for...[optional]
UserData  Object. Changes the instance's user data to the specified value.[optional]
Value
Character. A new value for the attribute. [optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response [optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.

region
Character. The region of the AWS service.

Value
A list object or a character vector

InstanceId
The ID of the instance.

SourceDestCheck
Specifies whether source/destination checking is enabled. A value of true means that checking is enabled, and false means that checking is disabled. This value must be false for a NAT instance to perform NAT.

Attribute
The name of the attribute.

BlockDeviceMapping
Modifies the DeleteOnTermination attribute for volumes that are currently attached. The volume must be owned by the caller. If no value is specified for DeleteOnTermination, the default is true and the volume is deleted when the instance is terminated.

To add instance store volumes to an Amazon EBS-backed instance, you must add them when you launch the instance. For more information, see Updating the block device mapping when launching an instance in the Amazon EC2 User Guide.

DisableApiTermination
If the value is true, you can\’t terminate the instance using the Amazon EC2 console, CLI, or API; otherwise, you can. You cannot use this parameter for Spot Instances.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

EbsOptimized

Specifies whether the instance is optimized for Amazon EBS I/O. This optimization provides dedicated throughput to Amazon EBS and an optimized configuration stack to provide optimal EBS I/O performance. This optimization isn’t available with all instance types. Additional usage charges apply when using an EBS Optimized instance.

EnaSupport

Set to true to enable enhanced networking with ENA for the instance.

This option is supported only for HVM instances. Specifying this option with a PV instance can make it unreachable.

GroupId

[EC2-VPC] Changes the security groups of the instance. You must specify at least one security group, even if it’s just the default security group for the VPC. You must specify the security group ID, not the security group name.

InstanceInitiatedShutdownBehavior

Specifies whether an instance stops or terminates when you initiate shutdown from the instance (using the operating system command for system shutdown).

InstanceType

Changes the instance type to the specified value. For more information, see Instance types in the Amazon EC2 User Guide. If the instance type is not valid, the error returned is InvalidInstanceAttributeValue.

Kernel

Changes the instance’s kernel to the specified value. We recommend that you use PV-GRUB instead of kernels and RAM disks. For more information, see PV-GRUB.

Ramdisk

Changes the instance’s RAM disk to the specified value. We recommend that you use PV-GRUB instead of kernels and RAM disks. For more information, see PV-GRUB.
SriovNetSupport

Set to simple to enable enhanced networking with the Intel 82599 Virtual Function interface for the instance.

There is no way to disable enhanced networking with the Intel 82599 Virtual Function interface at this time.

This option is supported only for HVM instances. Specifying this option with a PV instance can make it unreachable.

UserData

Changes the instance’s user data to the specified value. If you are using an AWS SDK or command line tool, base64-encoding is performed for you, and you can load the text from a file. Otherwise, you must provide base64-encoded text.

Value

A new value for the attribute. Use only with the kernel, ramdisk, userData, disableApiTermination, or instanceInitiatedShutdownBehavior attribute.

---

**ec2_modify_instance_capacity_reservation_attributes**

Modify Instance Capacity Reservation Attributes

---

Description

Modifies the Capacity Reservation settings for a stopped instance. Use this action to configure an instance to target a specific Capacity Reservation, run in any open Capacity Reservation with matching attributes, or run On-Demand Instance capacity.

Usage

```python
ec2_modify_instance_capacity_reservation_attributes(
    InstanceId,
    CapacityReservationSpecification,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)```
**ec2_modify_instance_credit_specification**

**Arguments**

- **InstanceId**  Character. The ID of the instance to be modified.
- **CapacityReservationSpecification**  Object. Information about the Capacity Reservation targeting option.
- **DryRun**  Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
- **simplify**  Logical. Whether to simplify the result and handle `nextToken` in the response[optional]
- **others**  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**  Logical. Whether to show an error message when a network error occurs.
- **retry_time**  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**  Character. The region of the AWS service.

**Value**

A list object or a character vector

**InstanceId**

The ID of the instance to be modified.

**CapacityReservationSpecification**

Information about the Capacity Reservation targeting option.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

---

**ec2_modify_instance_credit_specification**

*Modify Instance Credit Specification*

**Description**

Modify Instance Credit Specification
Usage

```r
e2_modify_instance_credit_specification(
    InstanceCreditSpecification,
    DryRun = NULL,
    ClientToken = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **InstanceCreditSpecification**  
  List. Information about the credit option for CPU usage.

- **DryRun**  
  Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]

- **ClientToken**  
  Character. A unique, case-sensitive token that you provide to ensure idempotency of your modification request. [optional]

- **simplify**  
  Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]

- **others**  
  Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]

- **print_on_error**  
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**  
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

- **network_timeout**  
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**  
  Character. The region of the AWS service.

Value

A list object or a character vector

**InstanceCreditSpecification**

Information about the credit option for CPU usage.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`. 
ClientToken

A unique, case-sensitive token that you provide to ensure idempotency of your modification request. For more information, see Ensuring Idempotency.

---

Modify Instance Event Start Time

Description

Modifies the start time for a scheduled Amazon EC2 instance event.

Usage

```
ec2_modify_instance_event_start_time(
    InstanceId,
    InstanceEventId,
    NotBefore,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **InstanceId**  Character. The ID of the instance with the scheduled event.
- **InstanceEventId**  Character. The ID of the event whose date and time you are modifying.
- **NotBefore**  Character. The new date and time when the event will take place.
- **DryRun**  Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **simplify**  Logical. Whether to simplify the result and handle nextToken in the response [optional]
- **others**  Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]
- **print_on_error**  Logical. Whether to show an error message when a network error occurs.
- **retry_time**  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- **network_timeout**  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**  Character. The region of the AWS service.
Value

A list object or a character vector

InstanceId

The ID of the instance with the scheduled event.

InstanceEventId

The ID of the event whose date and time you are modifying.

NotBefore

The new date and time when the event will take place.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**ec2_modify_instance_metadata_options**

*Modify Instance Metadata Options*

---

**Description**

Modify the instance metadata parameters on a running or stopped instance. When you modify the parameters on a stopped instance, they are applied when the instance is started. When you modify the parameters on a running instance, the API responds with a state of pending. After the parameter modifications are successfully applied to the instance, the state of the modifications changes from pending to applied in subsequent describe-instances API calls. For more information, see Instance metadata and user data in the Amazon EC2 User Guide.

**Usage**

```r
ec2_modify_instance_metadata_options(
  InstanceId,
  HttpTokens = NULL,
  HttpPutResponseHopLimit = NULL,
  HttpEndpoint = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
Arguments

- **InstanceId** Character. The ID of the instance.
- **HttpTokens** Character. The state of token usage for your instance metadata requests. [optional]
- **HttpPutResponseHopLimit** Integer. The desired HTTP PUT response hop limit for instance metadata requests. [optional]
- **HttpEndpoint** Character. This parameter enables or disables the HTTP metadata endpoint on your instances. [optional]
- **DryRun** Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **simplify** Logical. Whether to simplify the result and handle `nextToken` in the response [optional]
- **others** Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]
- **print_on_error** Logical. Whether to show an error message when a network error occurs.
- **retry_time** Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout** Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region** Character. The region of the AWS service.

Value

A list object or a character vector

**InstanceId**

The ID of the instance.

**HttpTokens**

The state of token usage for your instance metadata requests. If the parameter is not specified in the request, the default state is optional.

If the state is optional, you can choose to retrieve instance metadata with or without a signed token header on your request. If you retrieve the IAM role credentials without a token, the version 1.0 role credentials are returned. If you retrieve the IAM role credentials using a valid signed token, the version 2.0 role credentials are returned.

If the state is required, you must send a signed token header with any instance metadata retrieval requests. In this state, retrieving the IAM role credential always returns the version 2.0 credentials; the version 1.0 credentials are not available.
HttpPutResponseHopLimit

The desired HTTP PUT response hop limit for instance metadata requests. The larger the number, the further instance metadata requests can travel. If no parameter is specified, the existing state is maintained.

Possible values: Integers from 1 to 64

HttpEndpoint

This parameter enables or disables the HTTP metadata endpoint on your instances. If the parameter is not specified, the existing state is maintained.

If you specify a value of disabled, you will not be able to access your instance metadata.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_modify_instance_placement

Modify Instance Placement

Description

Modify Instance Placement

Usage

```r
ec2_modify_instance_placement(
  InstanceId,
  Affinity = NULL,
  GroupName = NULL,
  HostId = NULL,
  Tenancy = NULL,
  PartitionNumber = NULL,
  HostResourceGroupArn = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
Arguments

InstanceId  Character. The ID of the instance that you are modifying.
Affinity      Character. The affinity setting for the instance.[optional]
GroupName     Character. The name of the placement group in which to place the instance.[optional]
HostId        Character. The ID of the Dedicated Host with which to associate the instance.[optional]
Tenancy       Character. The tenancy for the instance.[optional]
PartitionNumber  Integer. Reserved for future use.[optional]
HostResourceGroupArn  Character. The ARN of the host resource group in which to place the instance.[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region       Character. The region of the AWS service.

Value

A list object or a character vector

InstanceId

The ID of the instance that you are modifying.

Affinity

The affinity setting for the instance.

GroupName

The name of the placement group in which to place the instance. For spread placement groups, the instance must have a tenancy of default. For cluster and partition placement groups, the instance must have a tenancy of default or dedicated.

To remove an instance from a placement group, specify an empty string ("\").

HostId

The ID of the Dedicated Host with which to associate the instance.
Tenancy

The tenancy for the instance.

PartitionNumber

Reserved for future use.

HostResourceGroupArn

The ARN of the host resource group in which to place the instance.

description

Modifies a launch template. You can specify which version of the launch template to set as the default version. When launching an instance, the default version applies when a launch template version is not specified.

Usage

e2_modify_launch_template(
  DryRun = NULL,
  ClientToken = NULL,
  LaunchTemplateId = NULL,
  LaunchTemplateName = NULL,
  SetDefaultVersion = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

ClientToken Character. Unique, case-sensitive identifier you provide to ensure the idempotency of the request.[optional]

LaunchTemplateId Character. The ID of the launch template.[optional]
LaunchTemplateName
Character. The name of the launch template.[optional]

SetDefaultVersion
Character. The version number of the launch template to set as the default version.[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value
A list object or a character vector

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ClientToken
Unique, case-sensitive identifier you provide to ensure the idempotency of the request. For more information, see Ensuring Idempotency.
Constraint: Maximum 128 ASCII characters.

LaunchTemplateId
The ID of the launch template. You must specify either the launch template ID or launch template name in the request.

LaunchTemplateName
The name of the launch template. You must specify either the launch template ID or launch template name in the request.

SetDefaultVersion
The version number of the launch template to set as the default version.
Modify Managed Prefix List

Usage

ec2_modify_managed_prefix_list(
    PrefixListId,
    DryRun = NULL,
    CurrentVersion = NULL,
    PrefixListName = NULL,
    AddEntry = NULL,
    RemoveEntry = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

PrefixListId Character. The ID of the prefix list.
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
CurrentVersion Integer. The current version of the prefix list.[optional]
PrefixListName Character. A name for the prefix list.[optional]
AddEntry List. One or more entries to add to the prefix list.[optional]
RemoveEntry List. One or more entries to remove from the prefix list.[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encountering the network issue. If the request has been sent retry_time times but still not able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.
region Character. The region of the AWS service.
Value
A list object or a character vector

PrefixListId
The ID of the prefix list.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

CurrentVersion
The current version of the prefix list.

PrefixListName
A name for the prefix list.

AddEntry
One or more entries to add to the prefix list.

RemoveEntry
One or more entries to remove from the prefix list.

---

**ec2_modify_network_interface_attribute**

*Modify Network Interface Attribute*

---

**Description**
Modifies the specified network interface attribute. You can specify only one attribute at a time. You can use this action to attach and detach security groups from an existing EC2 instance.

**Usage**

```r
ec2_modify_network_interface_attribute(
  NetworkInterfaceId,
  Attachment = NULL,
  Description = NULL,
  DryRun = NULL,
  SecurityGroupId = NULL,
  SourceDestCheck = NULL,
  simplify = TRUE,
)```
ec2_modify_network_interface_attribute

others = list(),
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

NetworkInterfaceId
    Character. The ID of the network interface.
Attachment
    Object. Information about the interface attachment.[optional]
Description
    Object. A description for the network interface.[optional]
DryRun
    Logical. Checks whether you have the required permissions for the action, without actually making the request,[optional]
SecurityGroupId
    List. Changes the security groups for the network interface.[optional]
SourceDestCheck
    Object. Indicates whether source/destination checking is enabled.[optional]
simplify
    Logical. Whether to simplify the result and handle nextToken in the response[optional]
others
    Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error
    Logical. Whether to show an error message when a network error occurs.
retry_time
    Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout
    Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region
    Character. The region of the AWS service.

Value

A list object or a character vector

NetworkInterfaceId
    The ID of the network interface.
Attachment
    Information about the interface attachment. If modifying the 'delete on termination' attribute, you must specify the ID of the interface attachment.
Description
    A description for the network interface.
DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunError. Otherwise, it is UnauthorizedOperation.

SecurityGroupId
Changes the security groups for the network interface. The new set of groups you specify replaces the current set. You must specify at least one group, even if it's just the default security group in the VPC. You must specify the ID of the security group, not the name.

SourceDestCheck
Indicates whether source/destination checking is enabled. A value of true means checking is enabled, and false means checking is disabled. This value must be false for a NAT instance to perform NAT. For more information, see NAT Instances in the Amazon Virtual Private Cloud User Guide.

---

ec2_modify_reserved_instances

Modify Reserved Instances

---

Description
Modify Reserved Instances

Usage

c2_modify_reserved_instances(
    ReservedInstancesId,
    ReservedInstancesConfigurationSetItemType,
    ClientToken = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

ReservedInstancesId
List. The IDs of the Reserved Instances to modify.

ReservedInstancesConfigurationSetItemType
List. The configuration settings for the Reserved Instances to modify.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ClientToken</td>
<td>Character. A unique, case-sensitive token you provide to ensure idempotency of your modification request. [optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response [optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>

**Value**

A list object or a character vector

**ReservedInstancesId**

The IDs of the Reserved Instances to modify.

**ReservedInstancesConfigurationSetItemType**

The configuration settings for the Reserved Instances to modify.

**ClientToken**

A unique, case-sensitive token you provide to ensure idempotency of your modification request. For more information, see [Ensuring Idempotency](#).

---

**Description**

Modify Snapshot Attribute
ec2_modify_snapshot_attribute

Usage

ec2_modify_snapshot_attribute(
    SnapshotId,
    Attribute = NULL,
    CreateVolumePermission = NULL,
    UserGroup = NULL,
    OperationType = NULL,
    UserId = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

SnapshotId  Character. The ID of the snapshot.
Attribute    Character. The snapshot attribute to modify. Only volume creation permissions
            can be modified.[optional]
CreateVolumePermission Object. A JSON representation of the snapshot attribute modification.[optional]
UserGroup    List. The group to modify for the snapshot.[optional]
OperationType Character. The type of operation to perform to the attribute.[optional]
UserId      List. The account ID to modify for the snapshot.[optional]
DryRun      Logical. Checks whether you have the required permissions for the action, with-
            out actually making the request,.[optional]
simplify    Logical. Whether to simplify the result and handle nextToken in the response[optional]
others      Named list. The parameters that are not included in the function parameters and
            need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time   Integer. Number of retries for a REST request when encounter the network
            issue. If the request has been sent retry_time times but still not be able to get
            the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can
            not be less than 1 ms.
region      Character. The region of the AWS service.

Value

A list object or a character vector
SnapshotId

The ID of the snapshot.

Attribute

The snapshot attribute to modify. Only volume creation permissions can be modified.

CreateVolumePermission

A JSON representation of the snapshot attribute modification.

UserGroup

The group to modify for the snapshot.

OperationType

The type of operation to perform to the attribute.

UserId

The account ID to modify for the snapshot.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

description

Modify Spot Fleet Request

Usage

c2_modify_spot_fleet_request(
    SpotFleetRequestId,
    ExcessCapacityTerminationPolicy = NULL,
    LaunchTemplateConfig = NULL,
    TargetCapacity = NULL,
    OnDemandTargetCapacity = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

SpotFleetRequestId
    Character. The ID of the Spot Fleet request.

ExcessCapacityTerminationPolicy
    Character. Indicates whether running Spot Instances should be terminated if the target capacity of the Spot Fleet request is decreased below the current size of the Spot Fleet.

LaunchTemplateConfig
    List. The launch template and overrides.[optional]

TargetCapacity
    Integer. The size of the fleet.[optional]

OnDemandTargetCapacity
    Integer. The number of On-Demand Instances in the fleet.[optional]

simplify
    Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
    Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
    Logical. Whether to show an error message when a network error occurs.

retry_time
    Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
    Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
    Character. The region of the AWS service.

Value

A list object or a character vector

SpotFleetRequestId
    The ID of the Spot Fleet request.

ExcessCapacityTerminationPolicy
    Indicates whether running Spot Instances should be terminated if the target capacity of the Spot Fleet request is decreased below the current size of the Spot Fleet.

LaunchTemplateConfig
    The launch template and overrides. You can only use this parameter if you specified a launch template (LaunchTemplateConfigs) in your Spot Fleet request. If you specified LaunchSpecifications in your Spot Fleet request, then omit this parameter.
**TargetCapacity**

The size of the fleet.

**OnDemandTargetCapacity**

The number of On-Demand Instances in the fleet.

---

**ec2_modify_subnet_attribute**

*Modify Subnet Attribute*

---

**Description**

Modifies a subnet attribute. You can only modify one attribute at a time.

**Usage**

```python
ec2_modify_subnet_attribute(
    SubnetId,
    AssignIpv6AddressOnCreation = NULL,
    MapPublicIpOnLaunch = NULL,
    MapCustomerOwnedIpOnLaunch = NULL,
    CustomerOwnedIpv4Pool = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

**Arguments**

- **SubnetId** Character. The ID of the subnet.
- **AssignIpv6AddressOnCreation** Object. Specify true to indicate that network interfaces created in the specified subnet should be assigned...[optional]
- **MapPublicIpOnLaunch** Object. Specify true to indicate that network interfaces attached to instances created in the specified...[optional]
- **MapCustomerOwnedIpOnLaunch** Object. Specify true to indicate that network interfaces attached to instances created in the specified...[optional]
- **CustomerOwnedIpv4Pool** Character. The customer-owned IPv4 address pool associated with the subnet.[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response [optional]

others Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.

Value

A list object or a character vector

SubnetId

The ID of the subnet.

AssignIpv6AddressOnCreation

Specify true to indicate that network interfaces created in the specified subnet should be assigned an IPv6 address. This includes a network interface that’s created when launching an instance into the subnet (the instance therefore receives an IPv6 address).

If you enable the IPv6 addressing feature for your subnet, your network interface or instance only receives an IPv6 address if it’s created using version 2016-11-15 or later of the Amazon EC2 API.

MapPublicIpOnLaunch

Specify true to indicate that network interfaces attached to instances created in the specified subnet should be assigned a public IPv4 address.

MapCustomerOwnedIpOnLaunch

Specify true to indicate that network interfaces attached to instances created in the specified subnet should be assigned a customer-owned IPv4 address.

When this value is true, you must specify the customer-owned IP pool using CustomerOwnedIpv4Pool.

CustomerOwnedIpv4Pool

The customer-owned IPv4 address pool associated with the subnet.

You must set this value when you specify true for MapCustomerOwnedIpOnLaunch.
ec2_modify_traffic_mirror_filter_network_services

Modify Traffic Mirror Filter Network Services

Description

Modify Traffic Mirror Filter Network Services

Usage

```r
ec2_modify_traffic_mirror_filter_network_services(
  TrafficMirrorFilterId,
  AddNetworkService = NULL,
  RemoveNetworkService = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **TrafficMirrorFilterId**: Character. The ID of the Traffic Mirror filter.
- **AddNetworkService**: List. The network service, for example Amazon DNS, that you want to mirror.[optional]
- **RemoveNetworkService**: List. The network service, for example Amazon DNS, that you no longer want to mirror.[optional]
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response[optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.
Value
A list object or a character vector

TrafficMirrorFilterId
The ID of the Traffic Mirror filter.

AddNetworkService
The network service, for example Amazon DNS, that you want to mirror.

RemoveNetworkService
The network service, for example Amazon DNS, that you no longer want to mirror.

DryRun
Checks whether you have the required permissions for the action, without actually making the
request, and provides an error response. If you have the required permissions, the error response is
DryRunOperation. Otherwise, it is UnauthorizedOperation.

Usage

```
ec2_modify_traffic_mirror_filter_rule(
    TrafficMirrorFilterRuleId,  
    TrafficDirection = NULL,   
    RuleNumber = NULL,        
    RuleAction = NULL,        
    DestinationPortRange = NULL, 
    SourcePortRange = NULL,   
    Protocol = NULL,          
    DestinationCidrBlock = NULL, 
    SourceCidrBlock = NULL,   
    Description = NULL,       
    RemoveField = NULL,       
    DryRun = NULL,            
    simplify = TRUE,          
    others = list(),          
    print_on_error = aws_get_print_on_error(),
```
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments
TrafficMirrorFilterRuleId
Character. The ID of the Traffic Mirror rule.
TrafficDirection
Character. The type of traffic (ingress \ egress) to assign to the rule.[optional]
RuleNumber
Integer. The number of the Traffic Mirror rule.[optional]
RuleAction
Character. The action to assign to the rule.[optional]
DestinationPortRange
Object. The destination ports that are associated with the Traffic Mirror rule.[optional]
SourcePortRange
Object. The port range to assign to the Traffic Mirror rule.[optional]
Protocol
Integer. The protocol, for example TCP, to assign to the Traffic Mirror rule.[optional]
DestinationCidrBlock
Character. The destination CIDR block to assign to the Traffic Mirror rule.[optional]
SourceCidrBlock
Character. The source CIDR block to assign to the Traffic Mirror rule.[optional]
Description
Character. The description to assign to the Traffic Mirror rule.[optional]
RemoveField
List. The properties that you want to remove from the Traffic Mirror filter rule.[optional]
DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]
simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]
others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error
Logical. Whether to show an error message when a network error occurs.
retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region
Character. The region of the AWS service.

Value
A list object or a character vector
TrafficMirrorFilterRuleId
The ID of the Traffic Mirror rule.
TrafficDirection
The type of traffic (ingress \| egress) to assign to the rule.

RuleNumber
The number of the Traffic Mirror rule. This number must be unique for each Traffic Mirror rule in a given direction. The rules are processed in ascending order by rule number.

RuleAction
The action to assign to the rule.

DestinationPortRange
The destination ports that are associated with the Traffic Mirror rule.

SourcePortRange
The port range to assign to the Traffic Mirror rule.

Protocol
The protocol, for example TCP, to assign to the Traffic Mirror rule.

DestinationCidrBlock
The destination CIDR block to assign to the Traffic Mirror rule.

SourceCidrBlock
The source CIDR block to assign to the Traffic Mirror rule.

Description
The description to assign to the Traffic Mirror rule.

RemoveField
The properties that you want to remove from the Traffic Mirror filter rule.
When you remove a property from a Traffic Mirror filter rule, the property is set to the default.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
ec2_modify_traffic_mirror_session

Modify Traffic Mirror Session

Description

Modifies a Traffic Mirror session.

Usage

```r
ec2_modify_traffic_mirror_session(
    TrafficMirrorSessionId,
    TrafficMirrorTargetId = NULL,
    TrafficMirrorFilterId = NULL,
    PacketLength = NULL,
    SessionNumber = NULL,
    VirtualNetworkId = NULL,
    Description = NULL,
    RemoveField = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **TrafficMirrorSessionId**
  Character. The ID of the Traffic Mirror session.

- **TrafficMirrorTargetId**
  Character. The Traffic Mirror target.[optional]

- **TrafficMirrorFilterId**
  Character. The ID of the Traffic Mirror filter.[optional]

- **PacketLength**
  Integer. The number of bytes in each packet to mirror.[optional]

- **SessionNumber**
  Integer. The session number determines the order in which sessions are evaluated when an interface is used.[optional]

- **VirtualNetworkId**
  Integer. The virtual network ID of the Traffic Mirror session.[optional]

- **Description**
  Character. The description to assign to the Traffic Mirror session.[optional]

- **RemoveField**
  List. The properties that you want to remove from the Traffic Mirror session.[optional]
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]

simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]

others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.

Value
A list object or a character vector

TrafficMirrorSessionId
The ID of the Traffic Mirror session.

TrafficMirrorTargetId
The Traffic Mirror target. The target must be in the same VPC as the source, or have a VPC peering connection with the source.

TrafficMirrorFilterId
The ID of the Traffic Mirror filter.

PacketLength
The number of bytes in each packet to mirror. These are bytes after the VXLAN header. To mirror a subset, set this to the length (in bytes) to mirror. For example, if you set this value to 100, then the first 100 bytes that meet the filter criteria are copied to the target. Do not specify this parameter when you want to mirror the entire packet.

SessionNumber
The session number determines the order in which sessions are evaluated when an interface is used by multiple sessions. The first session with a matching filter is the one that mirrors the packets. Valid values are 1-32766.

VirtualNetworkId
The virtual network ID of the Traffic Mirror session.
Description

The description to assign to the Traffic Mirror session.

RemoveField

The properties that you want to remove from the Traffic Mirror session.
When you remove a property from a Traffic Mirror session, the property is set to the default.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

---

```r
ec2_modify_transit_gateway

Modify Transit Gateway

Description

Modifies the specified transit gateway. When you modify a transit gateway, the modified options are applied to new transit gateway attachments only. Your existing transit gateway attachments are not modified.

Usage

```r
ec2_modify_transit_gateway(
    TransitGatewayId,
    Description = NULL,
    Options = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **TransitGatewayId**
  Character. The ID of the transit gateway.
- **Description**
  Character. The description for the transit gateway.[optional]
- **Options**
  Object. The options to modify.[optional]```
DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response [optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value
A list object or a character vector

TransitGatewayId
The ID of the transit gateway.

Description
The description for the transit gateway.

Options
The options to modify.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
Usage

```r
ec2_modify_transit_gateway_prefix_list_reference(
  TransitGatewayRouteTableId,
  PrefixListId,
  TransitGatewayAttachmentId = NULL,
  Blackhole = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **TransitGatewayRouteTableId**
  Character. The ID of the transit gateway route table.

- **PrefixListId**
  Character. The ID of the prefix list.

- **TransitGatewayAttachmentId**
  Character. The ID of the attachment to which traffic is routed.[optional]

- **Blackhole**
  Logical. Indicates whether to drop traffic that matches this route.[optional]

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  Character. The region of the AWS service.

Value

A list object or a character vector

**TransitGatewayRouteTableId**

The ID of the transit gateway route table.
PrefixListId

The ID of the prefix list.

TransitGatewayAttachmentId

The ID of the attachment to which traffic is routed.

Blackhole

Indicates whether to drop traffic that matches this route.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description

Modifies the specified VPC attachment.

Usage

ec2_modify_transit_gateway_vpc_attachment(
    TransitGatewayAttachmentId,
    AddSubnetIds = NULL,
    RemoveSubnetIds = NULL,
    Options = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

TransitGatewayAttachmentId
    Character. The ID of the attachment.

AddSubnetIds
    List. The IDs of one or more subnets to add. You can specify at most one subnet per Availability Zone.[optional]
RemoveSubnetIds
   List. The IDs of one or more subnets to remove.[optional]
Options
   Object. The new VPC attachment options. You cannot modify the IPv6 options.
      [optional]
DryRun
   Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
simplify
   Logical. Whether to simplify the result and handle nextToken in the response[optional]
others
   Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error
   Logical. Whether to show an error message when a network error occurs.
retry_time
   Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout
   Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region
   Character. The region of the AWS service.

Value
   A list object or a character vector

TransitGatewayAttachmentId
   The ID of the attachment.

AddSubnetIds
   The IDs of one or more subnets to add. You can specify at most one subnet per Availability Zone.

RemoveSubnetIds
   The IDs of one or more subnets to remove.

Options
   The new VPC attachment options.
   You cannot modify the IPv6 options.

DryRun
   Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
ec2_modify_volume

Modify Volume

Description
Modify Volume

Usage
ec2_modify_volume(
  VolumeId,
  DryRun = NULL,
  Size = NULL,
  VolumeType = NULL,
  Iops = NULL,
  Throughput = NULL,
  MultiAttachEnabled = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments
VolumeId Character. The ID of the volume.
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]
Size Integer. The target size of the volume, in GiB.[optional]
VolumeType Character. The target EBS volume type of the volume.[optional]
Iops Integer. The target IOPS rate of the volume.[optional]
Throughput Integer. The target throughput of the volume, in MiB/s.[optional]
MultiAttachEnabled Logical. Specifies whether to enable Amazon EBS Multi-Attach.[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout
   Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.

region
   Character. The region of the AWS service.

Value
   A list object or a character vector

VolumeId
   The ID of the volume.

DryRun
   Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Size
   The target size of the volume, in GiB. The target volume size must be greater than or equal to the existing size of the volume.
   The following are the supported volumes sizes for each volume type:
   - gp2 and gp3: 1-16,384
   - io1 and io2: 4-16,384
   - st1 and sc1: 125-16,384
   - standard: 1-1,024
   Default: If no size is specified, the existing size is retained.

VolumeType
   The target EBS volume type of the volume. For more information, see Amazon EBS volume types in the Amazon Elastic Compute Cloud User Guide.
   Default: If no type is specified, the existing type is retained.

Iops
   The target IOPS rate of the volume. This parameter is valid only for gp3, io1, and io2 volumes.
   The following are the supported values for each volume type:
   - gp3: 3,000-16,000 IOPS
   - io1: 100-64,000 IOPS
   - io2: 100-64,000 IOPS
   Default: If no IOPS value is specified, the existing value is retained.
Throughput

The target throughput of the volume, in MiB/s. This parameter is valid only for gp3 volumes. The maximum value is 1,000.

Default: If no throughput value is specified, the existing value is retained.
Valid Range: Minimum value of 125. Maximum value of 1000.

MultiAttachEnabled

Specifies whether to enable Amazon EBS Multi-Attach. If you enable Multi-Attach, you can attach the volume to up to 16 Nitro-based instances in the same Availability Zone. This parameter is supported with io1 and io2 volumes only. For more information, see Amazon EBS Multi-Attach in the Amazon Elastic Compute Cloud User Guide.

derive modify_volume_attribute

Modify Volume Attribute

Description

Modify Volume Attribute

Usage

ec2_modify_volume_attribute(
  VolumeId,
  AutoEnableIO = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

VolumeId Character. The ID of the volume.
AutoEnableIO Object. Indicates whether the volume should be auto-enabled for I/O operations.[optional]
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value
A list object or a character vector

VolumeId
The ID of the volume.

AutoEnableIO
Indicates whether the volume should be auto-enabled for I/O operations.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

ec2_modify_vpc_attribute

Modify Vpc Attribute

Description
Modifies the specified attribute of the specified VPC.

Usage

```r
ec2_modify_vpc_attribute(
  VpcId,
  EnableDnsHostnames = NULL,
  EnableDnsSupport = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
Arguments

VpcId  Character. The ID of the VPC.
EnableDnsHostnames  Object. Indicates whether the instances launched in the VPC get DNS hostnames.[optional]
EnableDnsSupport  Object. Indicates whether the DNS resolution is supported for the VPC.[optional]
simplify  Logical. Whether to simplify the result and handle nextToken in the response[optional]
others  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error  Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region  Character. The region of the AWS service.

Value

A list object or a character vector

VpcId

The ID of the VPC.

EnableDnsHostnames

Indicates whether the instances launched in the VPC get DNS hostnames. If enabled, instances in the VPC get DNS hostnames; otherwise, they do not.

You cannot modify the DNS resolution and DNS hostnames attributes in the same request. Use separate requests for each attribute. You can only enable DNS hostnames if you’ve enabled DNS support.

EnableDnsSupport

Indicates whether the DNS resolution is supported for the VPC. If enabled, queries to the Amazon provided DNS server at the 169.254.169.253 IP address, or the reserved IP address at the base of the VPC network range \' plus two\' succeed. If disabled, the Amazon provided DNS service in the VPC that resolves public DNS hostnames to IP addresses is not enabled.

You cannot modify the DNS resolution and DNS hostnames attributes in the same request. Use separate requests for each attribute.
ec2_modify_vpc_endpoint

Modify Vpc Endpoint

Description

Modifies attributes of a specified VPC endpoint. The attributes that you can modify depend on the type of VPC endpoint (interface, gateway, or Gateway Load Balancer). For more information, see VPC Endpoints in the Amazon Virtual Private Cloud User Guide.

Usage

ec2_modify_vpc_endpoint(
    VpcEndpointId,      
    DryRun = NULL,      
    ResetPolicy = NULL, 
    PolicyDocument = NULL, 
    AddRouteTableId = NULL,  
    RemoveRouteTableId = NULL,  
    AddSubnetId = NULL,  
    RemoveSubnetId = NULL,  
    AddSecurityGroupId = NULL,  
    RemoveSecurityGroupId = NULL,  
    PrivateDnsEnabled = NULL,  
    simplify = TRUE,  
    others = list(),  
    print_on_error = aws_get_print_on_error(),  
    retry_time = aws_get_retry_time(),  
    network_timeout = aws_get_network_timeout(),  
    region = aws_get_region()
)

Arguments

VpcEndpointId    Character. The ID of the endpoint.
DryRun          Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]
ResetPolicy     Logical. (Gateway endpoint) Specify true to reset the policy document to the default policy.[optional]
PolicyDocument  Character. (Interface and gateway endpoints) A policy to attach to the endpoint that controls access to the...[optional]
AddRouteTableId List. (Gateway endpoint) One or more route tables IDs to associate with the endpoint.[optional]
RemoveRouteTableId
   List. (Gateway endpoint) One or more route table IDs to disassociate from the endpoint.[optional]

AddSubnetId
   List. (Interface and Gateway Load Balancer endpoints) One or more subnet IDs in which to serve the endpoint.[optional]

RemoveSubnetId
   List. (Interface endpoint) One or more subnets IDs in which to remove the endpoint.[optional]

AddSecurityGroupId
   List. (Interface endpoint) One or more security group IDs to associate with the network interface.[optional]

RemoveSecurityGroupId
   List. (Interface endpoint) One or more security group IDs to disassociate from the network interface.[optional]

PrivateDnsEnabled
   Logical. (Interface endpoint) Indicates whether a private hosted zone is associated with the VPC.[optional]

simplify
   Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
   Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
   Logical. Whether to show an error message when a network error occurs.

retry_time
   Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
   Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
   Character. The region of the AWS service.

Value
   A list object or a character vector

VpcEndpointId
   The ID of the endpoint.

DryRun
   Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ResetPolicy
   (Gateway endpoint) Specify true to reset the policy document to the default policy. The default policy allows full access to the service.
PolicyDocument

(Interface and gateway endpoints) A policy to attach to the endpoint that controls access to the service. The policy must be in valid JSON format.

AddRouteTableId

(Gateway endpoint) One or more route tables IDs to associate with the endpoint.

RemoveRouteTableId

(Gateway endpoint) One or more route table IDs to disassociate from the endpoint.

AddSubnetId

(Interface and Gateway Load Balancer endpoints) One or more subnet IDs in which to serve the endpoint. For a Gateway Load Balancer endpoint, you can specify only one subnet.

RemoveSubnetId

(Interface endpoint) One or more subnets IDs in which to remove the endpoint.

AddSecurityGroupId

(Interface endpoint) One or more security group IDs to associate with the network interface.

RemoveSecurityGroupId

(Interface endpoint) One or more security group IDs to disassociate from the network interface.

PrivateDnsEnabled

(Interface endpoint) Indicates whether a private hosted zone is associated with the VPC.

Description

Modifies a connection notification for VPC endpoint or VPC endpoint service. You can change the SNS topic for the notification, or the events for which to be notified.
Usage

est2_modify_vpc_endpoint_connection_notification(
    ConnectionNotificationId,
    DryRun = NULL,
    ConnectionNotificationArn = NULL,
    ConnectionEvents = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

ConnectionNotificationId
    Character. The ID of the notification.

DryRun
    Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

ConnectionNotificationArn
    Character. The ARN for the SNS topic for the notification.[optional]

ConnectionEvents
    List. One or more events for the endpoint. Valid values are Accept, Connect, Delete, and Reject.[optional]

simplify
    Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
    Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
    Logical. Whether to show an error message when a network error occurs.

retry_time
    Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
    Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
    Character. The region of the AWS service.

Value

A list object or a character vector

ConnectionNotificationId

The ID of the notification.
Description

Modify Vpc Endpoint Service Configuration

Usage

```r
ec2_modify_vpc_endpoint_service_configuration(
    ServiceId,
    DryRun = NULL,
    PrivateDnsName = NULL,
    RemovePrivateDnsName = NULL,
    AcceptanceRequired = NULL,
    AddNetworkLoadBalancerArn = NULL,
    RemoveNetworkLoadBalancerArn = NULL,
    AddGatewayLoadBalancerArn = NULL,
    RemoveGatewayLoadBalancerArn = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **ServiceId**  Character. The ID of the service.
- **DryRun**     Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]
PrivateDnsName  Character. (Interface endpoint configuration) The private DNS name to assign to the endpoint service.[optional]

RemovePrivateDnsName  Logical. (Interface endpoint configuration) Removes the private DNS name of the endpoint service.[optional]

AcceptanceRequired  Logical. Indicates whether requests to create an endpoint to your service must be accepted.[optional]

AddNetworkLoadBalancerArn  List. The Amazon Resource Names (ARNs) of Network Load Balancers to add to your service configuration.[optional]

RemoveNetworkLoadBalancerArn  List. The Amazon Resource Names (ARNs) of Network Load Balancers to remove from your service configuration....[optional]

AddGatewayLoadBalancerArn  List. The Amazon Resource Names (ARNs) of Gateway Load Balancers to add to your service configuration.[optional]

RemoveGatewayLoadBalancerArn  List. The Amazon Resource Names (ARNs) of Gateway Load Balancers to remove from your service configuration....[optional]

simplify  Logical. Whether to simplify the result and handle nextToken in the response[optional]

others  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error  Logical. Whether to show an error message when a network error occurs.

retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  Character. The region of the AWS service.

Value

A list object or a character vector

ServiceId

The ID of the service.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
PrivateDnsName

(Interface endpoint configuration) The private DNS name to assign to the endpoint service.

RemovePrivateDnsName

(Interface endpoint configuration) Removes the private DNS name of the endpoint service.

AcceptanceRequired

Indicates whether requests to create an endpoint to your service must be accepted.

AddNetworkLoadBalancerArn

The Amazon Resource Names (ARNs) of Network Load Balancers to add to your service configuration.

RemoveNetworkLoadBalancerArn

The Amazon Resource Names (ARNs) of Network Load Balancers to remove from your service configuration.

AddGatewayLoadBalancerArn

The Amazon Resource Names (ARNs) of Gateway Load Balancers to add to your service configuration.

RemoveGatewayLoadBalancerArn

The Amazon Resource Names (ARNs) of Gateway Load Balancers to remove from your service configuration.

---

```r
ec2_modify_vpc_endpoint_service_permissions

Modify Vpc Endpoint Service Permissions

---

Description

Modify Vpc Endpoint Service Permissions

Usage

ext2_modify_vpc_endpoint_service_permissions(
  ServiceId,
  DryRun = NULL,
  AddAllowedPrincipals = NULL,
  RemoveAllowedPrincipals = NULL,
  simplify = TRUE,
  others = list(),
)```
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

ServiceId Character. The ID of the service.
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
AddAllowedPrincipals List. The Amazon Resource Names (ARN) of one or more principals.[optional]
RemoveAllowedPrincipals List. The Amazon Resource Names (ARN) of one or more principals.[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value

A list object or a character vector

ServiceId

The ID of the service.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

AddAllowedPrincipals

The Amazon Resource Names (ARN) of one or more principals. Permissions are granted to the principals in this list. To grant permissions to all principals, specify an asterisk (*).
RemoveAllowedPrincipals

The Amazon Resource Names (ARN) of one or more principals. Permissions are revoked for principals in this list.

---

description

Modify Vpc Peering Connection Options

Usage

```
ec2_modify_vpc_peering_connection_options(
    VpcPeeringConnectionId,
    AccepterPeeringConnectionOptions = NULL,
    DryRun = NULL,
    RequesterPeeringConnectionOptions = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **VpcPeeringConnectionId**
  Character. The ID of the VPC peering connection.

- **AccepterPeeringConnectionOptions**
  Object. The VPC peering connection options for the accepter VPC.[optional]

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

- **RequesterPeeringConnectionOptions**
  Object. The VPC peering connection options for the requester VPC.[optional]

- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
ec2_modify_vpc_tenancy

network_timeout
   Numeric. Number of seconds to wait for a REST response until giving up. Can
   not be less than 1 ms.

region
   Character. The region of the AWS service.

Value
   A list object or a character vector

VpcPeeringConnectionId
   The ID of the VPC peering connection.

AccepterPeeringConnectionOptions
   The VPC peering connection options for the accepter VPC.

DryRun
   Checks whether you have the required permissions for the action, without actually making the
   request, and provides an error response. If you have the required permissions, the error response is
   DryRunOperation. Otherwise, it is UnauthorizedOperation.

RequesterPeeringConnectionOptions
   The VPC peering connection options for the requester VPC.

ec2_modify_vpc_tenancy
   Modify Vpc Tenancy

Description
   Modify Vpc Tenancy

Usage
   ec2_modify_vpc_tenancy(
      VpcId,
      InstanceTenancy,
      DryRun = NULL,
      simplify = TRUE,
      others = list(),
      print_on_error = aws_get_print_on_error(),
      retry_time = aws_get_retry_time(),
      network_timeout = aws_get_network_timeout(),
      region = aws_get_region()
   )
ec2_modify_vpn_connection

**Arguments**

- **VpcId**  
  Character. The ID of the VPC.

- **InstanceTenancy**  
  Character. The instance tenancy attribute for the VPC.

- **DryRun**  
  Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]

- **Simplify**  
  Logical. Whether to simplify the result and handle `nextToken` in the response[optional]

- **Others**  
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **PrintOnError**  
  Logical. Whether to show an error message when a network error occurs.

- **RetryTime**  
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

- **NetworkTimeout**  
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **Region**  
  Character. The region of the AWS service.

**Value**

A list object or a character vector

**VpcId**

The ID of the VPC.

**InstanceTenancy**

The instance tenancy attribute for the VPC.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

---

**Modify Vpn Connection**

**Description**

Modify Vpn Connection
Usage

```r
ec2_modify_vpn_connection(
  VpnConnectionId,
  TransitGatewayId = NULL,
  CustomerGatewayId = NULL,
  VpnGatewayId = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **VpnConnectionId**
  Character. The ID of the VPN connection.

- **TransitGatewayId**
  Character. The ID of the transit gateway.[optional]

- **CustomerGatewayId**
  Character. The ID of the customer gateway at your end of the VPN connection.[optional]

- **VpnGatewayId**
  Character. The ID of the virtual private gateway at the AWS side of the VPN connection.[optional]

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]

- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  Character. The region of the AWS service.

Value

- A list object or a character vector

**VpnConnectionId**

The ID of the VPN connection.
TransitGatewayId

The ID of the transit gateway.

CustomerGatewayId

The ID of the customer gateway at your end of the VPN connection.

VpnGatewayId

The ID of the virtual private gateway at the AWS side of the VPN connection.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

```r
ec2_modify_vpn_connection_options

Modify Vpn Connection Options

description

Modify Vpn Connection Options

usage

c2.modify_vpn_connection_options(
  VpnConnectionId,
  LocalIpv4NetworkCidr = NULL,
  RemoteIpv4NetworkCidr = NULL,
  LocalIpv6NetworkCidr = NULL,
  RemoteIpv6NetworkCidr = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)```
Arguments

VpnConnectionId
Character. The ID of the Site-to-Site VPN connection.

LocalIpv4NetworkCidr
Character. The IPv4 CIDR on the customer gateway (on-premises) side of the VPN connection.[optional]

RemoteIpv4NetworkCidr
Character. The IPv4 CIDR on the AWS side of the VPN connection. Default: 0.0.0.0/0 [optional]

LocalIpv6NetworkCidr
Character. The IPv6 CIDR on the customer gateway (on-premises) side of the VPN connection.[optional]

RemoteIpv6NetworkCidr
Character. The IPv6 CIDR on the AWS side of the VPN connection. Default: ::/0 [optional]

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value

A list object or a character vector

VpnConnectionId
The ID of the Site-to-Site VPN connection.

LocalIpv4NetworkCidr
The IPv4 CIDR on the customer gateway (on-premises) side of the VPN connection.
Default: 0.0.0.0/0

RemoteIpv4NetworkCidr
The IPv4 CIDR on the AWS side of the VPN connection.
Default: 0.0.0.0/0
LocalIpv6NetworkCidr

The IPv6 CIDR on the customer gateway (on-premises) side of the VPN connection.
Default: ::/0

RemoteIpv6NetworkCidr

The IPv6 CIDR on the AWS side of the VPN connection.
Default: ::/0

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_modify_vpn_tunnel_certificate

Modify Vpn Tunnel Certificate

Description

Modifies the VPN tunnel endpoint certificate.

Usage

ec2_modify_vpn_tunnel_certificate(
  VpnConnectionId,
  VpnTunnelOutsideIpAddress,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

VpnConnectionId
  Character. The ID of the AWS Site-to-Site VPN connection.

VpnTunnelOutsideIpAddress
  Character. The external IP address of the VPN tunnel.

DryRun
  Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

simplify
  Logical. Whether to simplify the result and handle nextToken in the response[optional]
ec2_modify_vpn_tunnel_options

others  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error  Logical. Whether to show an error message when a network error occurs.

retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  Character. The region of the AWS service.

Value

A list object or a character vector

VpnConnectionId

The ID of the AWS Site-to-Site VPN connection.

VpnTunnelOutsideIpAddress

The external IP address of the VPN tunnel.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description

Modifies the options for a VPN tunnel in an AWS Site-to-Site VPN connection. You can modify multiple options for a tunnel in a single request, but you can only modify one tunnel at a time. For more information, see Site-to-Site VPN Tunnel Options for Your Site-to-Site VPN Connection in the AWS Site-to-Site VPN User Guide.
Usage

ee2_modify_vpn_tunnel_options(
    VpnConnectionId,
    VpnTunnelOutsideIpAddress,
    TunnelOptions,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

VpnConnectionId
   Character. The ID of the AWS Site-to-Site VPN connection.
VpnTunnelOutsideIpAddress
   Character. The external IP address of the VPN tunnel.
TunnelOptions
   Object. The tunnel options to modify.
DryRun
   Logical. Checks whether you have the required permissions for the action without actually making the request...[optional]
simplify
   Logical. Whether to simplify the result and handle nextToken in the response[optional]
others
   Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error
   Logical. Whether to show an error message when a network error occurs.
retry_time
   Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout
   Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region
   Character. The region of the AWS service.

Value

A list object or a character vector

VpnConnectionId

The ID of the AWS Site-to-Site VPN connection.

VpnTunnelOutsideIpAddress

The external IP address of the VPN tunnel.
**TunnelOptions**

The tunnel options to modify.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

```r
ec2_monitor_instances(InstanceId,
DryRun = NULL,
simplify = TRUE,
others = list(),
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>InstanceId</td>
<td>List. The IDs of the instances.</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response. [optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>
**Value**

A list object or a character vector

**InstanceIds**

The IDs of the instances.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

---

**Usage**

```r
ec2_move_address_to_vpc(
  PublicIp,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- **PublicIp**: Character. The Elastic IP address.
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response[optional]
ec2_provision_byoip_cidr

Provision Byoip Cidr

Description

Provision Byoip Cidr

Usage

```r
ec2_provision_byoip_cidr(
  Cidr,
  CidrAuthorizationContext = NULL,
  PubliclyAdvertisable = NULL,
  Description = NULL,
  DryRun = NULL,
  PoolTagSpecification = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
```
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

Cidr Character. The public IPv4 or IPv6 address range, in CIDR notation.

CidrAuthorizationContext Object. A signed document that proves that you are authorized to bring the specified IP address range to...[optional]

PubliclyAdvertisable Logical. (IPv6 only) Indicate whether the address range will be publicly advertised to the internet.[optional]

Description Character. A description for the address range and the address pool.[optional]

DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

PoolTagSpecification List. The tags to apply to the address pool.[optional]

simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]

others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.

Value

A list object or a character vector

Cidr

The public IPv4 or IPv6 address range, in CIDR notation. The most specific IPv4 prefix that you can specify is /24. The most specific IPv6 prefix you can specify is /56. The address range cannot overlap with another address range that you've brought to this or another Region.

CidrAuthorizationContext

A signed document that proves that you are authorized to bring the specified IP address range to Amazon using BYOIP.
PubliclyAdvertisable

(IPv6 only) Indicate whether the address range will be publicly advertised to the internet.
Default: true

Description

A description for the address range and the address pool.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

PoolTagSpecification

The tags to apply to the address pool.

---

tag

---

Description

Purchase a reservation with configurations that match those of your Dedicated Host. You must have active Dedicated Hosts in your account before you purchase a reservation. This action results in the specified reservation being purchased and charged to your account.

Usage

c2_purchase_host_reservation(
    HostIdSet,
    OfferingId,
    ClientToken = NULL,
    CurrencyCode = NULL,
    LimitPrice = NULL,
    TagSpecification = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region())
Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HostIdSet</td>
<td>List. The IDs of the Dedicated Hosts with which the reservation will be associated.</td>
</tr>
<tr>
<td>OfferingId</td>
<td>Character. The ID of the offering.</td>
</tr>
<tr>
<td>ClientToken</td>
<td>Character. Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. [optional]</td>
</tr>
<tr>
<td>CurrencyCode</td>
<td>Character. The currency in which the totalUpfrontPrice, LimitPrice, and totalHourlyPrice amounts are specified. [optional]</td>
</tr>
<tr>
<td>LimitPrice</td>
<td>Character. The specified limit is checked against the total upfront cost of the reservation (calculated as the offering's upfront cost multiplied by the host count). [optional]</td>
</tr>
<tr>
<td>TagSpecification</td>
<td>List. The tags to apply to the Dedicated Host Reservation during purchase. [optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response. [optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>

Value

A list object or a character vector

HostIdSet

The IDs of the Dedicated Hosts with which the reservation will be associated.

OfferingId

The ID of the offering.

ClientToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. For more information, see Ensuring Idempotency.

CurrencyCode

The currency in which the totalUpfrontPrice, LimitPrice, and totalHourlyPrice amounts are specified. At this time, the only supported currency is USD.
LimitPrice

The specified limit is checked against the total upfront cost of the reservation (calculated as the offering's upfront cost multiplied by the host count). If the total upfront cost is greater than the specified price limit, the request fails. This is used to ensure that the purchase does not exceed the expected upfront cost of the purchase. At this time, the only supported currency is USD. For example, to indicate a limit price of USD 100, specify 100.00.

TagSpecification

The tags to apply to the Dedicated Host Reservation during purchase.

description

Purchase Reserved Instances Offering

Usage

c2_purchase_reserved_instances_offering(
    InstanceCount,
    ReservedInstancesOfferingId,
    DryRun = NULL,
    LimitPrice = NULL,
    PurchaseTime = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

InstanceCount  Integer. The number of Reserved Instances to purchase.
ReservedInstancesOfferingId  Character. The ID of the Reserved Instance offering to purchase.
DryRun  Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
LimitPrice  Object. Specified for Reserved Instance Marketplace offerings to limit the total order and ensure that the...[optional]
PurchaseTime  Character. The time at which to purchase the Reserved Instance, in UTC format (for example, YYYY-MM-DDTHH:MM:SSZ)....[optional]
**Value**

A list object or a character vector

**InstanceCount**

The number of Reserved Instances to purchase.

**ReservedInstancesOfferingId**

The ID of the Reserved Instance offering to purchase.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

**LimitPrice**

Specified for Reserved Instance Marketplace offerings to limit the total order and ensure that the Reserved Instances are not purchased at unexpected prices.

**PurchaseTime**

The time at which to purchase the Reserved Instance, in UTC format (for example, `YYYY-MM-DDTHH:MM:SSZ`).
ec2_purchase_scheduled_instances

Purchase Scheduled Instances

Description
Purchase Scheduled Instances

Usage

```r
ec2_purchase_scheduled_instances(
  PurchaseRequest,
  ClientToken = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PurchaseRequest</td>
<td>List. The purchase requests.</td>
</tr>
<tr>
<td>ClientToken</td>
<td>Character. Unique, case-sensitive identifier that ensures the idempotency of the request. [optional]</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response. [optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>

Value

A list object or a character vector
**PurchaseRequest**

The purchase requests.

**ClientToken**

Unique, case-sensitive identifier that ensures the idempotency of the request. For more information, see [Ensuring Idempotency](#).

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

---

**ec2_reboot_instances**  
*Reboot Instances*

**Description**

Reboot Instances

**Usage**

```r
ec2_reboot_instances(
  InstanceId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- **InstanceId**  
  List. The instance IDs.
- **DryRun**  
  Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]
- **simplify**  
  Logical. Whether to simplify the result and handle `nextToken` in the response[optional]
- **others**  
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**  
  Logical. Whether to show an error message when a network error occurs.
- **retry_time**  
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
network_timeout
   Numeric. Number of seconds to wait for a REST response until giving up. Can
   not be less than 1 ms.
region
   Character. The region of the AWS service.

Value
   A list object or a character vector

InstanceId
   The instance IDs.

DryRun
   Checks whether you have the required permissions for the action, without actually making the
   request, and provides an error response. If you have the required permissions, the error response is
   DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description
   Register Image

Usage
   ec2_register_image(
     Name,
     ImageLocation = NULL,
     Architecture = NULL,
     BlockDeviceMapping = NULL,
     Description = NULL,
     DryRun = NULL,
     EnaSupport = NULL,
     KernelId = NULL,
     BillingProduct = NULL,
     RamdiskId = NULL,
     RootDeviceName = NULL,
     SriovNetSupport = NULL,
     VirtualizationType = NULL,
     simplify = TRUE,
     others = list(),
     print_on_error = aws_get_print_on_error(),
     retry_time = aws_get_retry_time(),
     network_timeout = aws_get_network_timeout(),
     region = aws_get_region()
   )
Arguments

- **Name**: Character. A name for your AMI.
- **ImageLocation**: Character. The full path to your AMI manifest in Amazon S3 storage. [optional]
- **Architecture**: Character. The architecture of the AMI. [optional]
- **BlockDeviceMapping**: List. The block device mapping entries. [optional]
- **Description**: Character. A description for your AMI. [optional]
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **EnaSupport**: Logical. Set to true to enable enhanced networking with ENA for the AMI and any instances that you launch. [optional]
- ** KernelId**: Character. The ID of the kernel. [optional]
- **BillingProduct**: List. The billing product codes. [optional]
- **RamdiskId**: Character. The ID of the RAM disk. [optional]
- **RootDeviceName**: Character. The device name of the root device volume (for example, /dev/sda1). [optional]
- **SriovNetSupport**: Character. Set to simple to enable enhanced networking with the Intel 82599 Virtual Function interface for... [optional]
- **VirtualizationType**: Character. The type of virtualization (hvm | paravirtual). Default: paravirtual [optional]
- **simplify**: Logical. Whether to simplify the result and handle nextToken in the response. [optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

Name

A name for your AMI.

Constraints: 3-128 alphanumeric characters, parentheses (()), square brackets ([]), spaces ( ), periods (.), slashes (/), dashes (-), single quotes (‘), at-signs (@), or underscores (_)
ImageLocation

The full path to your AMI manifest in Amazon S3 storage. The specified bucket must have the `aws-exec-read` canned access control list (ACL) to ensure that it can be accessed by Amazon EC2. For more information, see Canned ACLs in the Amazon S3 Service Developer Guide.

Architecture

The architecture of the AMI.

Default: For Amazon EBS-backed AMIs, i386. For instance store-backed AMIs, the architecture specified in the manifest file.

BlockDeviceMapping

The block device mapping entries.

If you specify an EBS volume using the ID of an EBS snapshot, you can’t specify the encryption state of the volume.

If you create an AMI on an Outpost, then all backing snapshots must be on the same Outpost or in the Region of that Outpost. AMIs on an Outpost that include local snapshots can be used to launch instances on the same Outpost only. For more information, Amazon EBS local snapshots on Outposts in the Amazon Elastic Compute Cloud User Guide.

Description

A description for your AMI.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

EnaSupport

Set to true to enable enhanced networking with ENA for the AMI and any instances that you launch from the AMI.

This option is supported only for HVM AMIs. Specifying this option with a PV AMI can make instances launched from the AMI unreachable.

KernelId

The ID of the kernel.

BillingProduct

The billing product codes. Your account must be authorized to specify billing product codes. Otherwise, you can use the AWS Marketplace to bill for the use of an AMI.
RamdiskId

The ID of the RAM disk.

RootDeviceName

The device name of the root device volume (for example, /dev/sda1).

SriovNetSupport

Set to simple to enable enhanced networking with the Intel 82599 Virtual Function interface for the AMI and any instances that you launch from the AMI.

There is no way to disable sriovNetSupport at this time.

This option is supported only for HVM AMIs. Specifying this option with a PV AMI can make instances launched from the AMI unreachable.

VirtualizationType

The type of virtualization (hvm \ paravirtual).

Default: paravirtual

---

ec2_register_instance_event_notification_attributes

Register Instance Event Notification Attributes

---

Description

Register Instance Event Notification Attributes

Usage

```python
ec2_register_instance_event_notification_attributes(
    DryRun = NULL,
    InstanceTagAttribute = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```
**Arguments**

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request, ...[optional]

- **InstanceTagAttribute**
  Object. Information about the tag keys to register.[optional]

- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  Character. The region of the AWS service.

**Value**

A list object or a character vector

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

**InstanceTagAttribute**

Information about the tag keys to register.
Usage

```r
ec2_register_transit_gateway_multicast_group_members(
  TransitGatewayMulticastDomainId = NULL,
  GroupIpAddress = NULL,
  NetworkInterfaceIds = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **TransitGatewayMulticastDomainId**
  Character. The ID of the transit gateway multicast domain. [optional]

- **GroupIpAddress**
  Character. The IP address assigned to the transit gateway multicast group. [optional]

- **NetworkInterfaceIds**
  List. The group members’ network interface IDs to register with the transit gateway multicast group. [optional]

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]

- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  Character. The region of the AWS service.

Value

A list object or a character vector

**TransitGatewayMulticastDomainId**

The ID of the transit gateway multicast domain.

**GroupIpAddress**

The IP address assigned to the transit gateway multicast group.
NetworkInterfaceIds

The group members' network interface IDs to register with the transit gateway multicast group.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_register_transit_gateway_multicast_group_sources
Register Transit Gateway Multicast Group Sources

Description

Register Transit Gateway Multicast Group Sources

Usage

ec2_register_transit_gateway_multicast_group_sources(
    TransitGatewayMulticastDomainId = NULL,
    GroupIpAddress = NULL,
    NetworkInterfaceIds = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

TransitGatewayMulticastDomainId
    Character. The ID of the transit gateway multicast domain.[optional]

GroupIpAddress
    Character. The IP address assigned to the transit gateway multicast group.[optional]

NetworkInterfaceIds
    List. The group members' network interface IDs to register with the transit gateway multicast group.[optional]

DryRun
    Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]

simplify
    Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
    Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error  Logical. Whether to show an error message when a network error occurs.
retry_time   Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region   Character. The region of the AWS service.

Value

A list object or a character vector

TransitGatewayMulticastDomainId

The ID of the transit gateway multicast domain.

GroupIpAddress

The IP address assigned to the transit gateway multicast group.

NetworkInterfaceIds

The group sources\ network interface IDs to register with the transit gateway multicast group.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_reject_transit_gateway_multicast_domain_associations

Reject Transit Gateway Multicast Domain Associations

Description

Rejects a request to associate cross-account subnets with a transit gateway multicast domain.

Usage

ec2_reject_transit_gateway_multicast_domain_associations(
    TransitGatewayMulticastDomainId = NULL,
    TransitGatewayAttachmentId = NULL,
    SubnetIds = NULL,
    DryRun = NULL,
    simplify = TRUE,
others = list(),
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

TransitGatewayMulticastDomainId
   Character. The ID of the transit gateway multicast domain.[optional]

TransitGatewayAttachmentId
   Character. The ID of the transit gateway attachment.[optional]

SubnetIds
   List. The IDs of the subnets to associate with the transit gateway multicast domain.[optional]

DryRun
   Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]

simplify
   Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
   Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
   Logical. Whether to show an error message when a network error occurs.

retry_time
   Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
   Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
   Character. The region of the AWS service.

Value

A list object or a character vector

TransitGatewayMulticastDomainId

The ID of the transit gateway multicast domain.

TransitGatewayAttachmentId

The ID of the transit gateway attachment.

SubnetIds

The IDs of the subnets to associate with the transit gateway multicast domain.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

ec2_reject_transit_gateway_peering_attachment

*Reject Transit Gateway Peering Attachment*

---

**Description**

Rejects a transit gateway peering attachment request.

**Usage**

```r
ec2_reject_transit_gateway_peering_attachment(
    TransitGatewayAttachmentId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

**Arguments**

- **TransitGatewayAttachmentId**
  Character. The ID of the transit gateway peering attachment.

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]

- **simplify**
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  Character. The region of the AWS service.
Value

A list object or a character vector

TransitGatewayAttachmentId

The ID of the transit gateway peering attachment.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_reject_transit_gateway_vpc_attachment

Reject Transit Gateway Vpc Attachment

Description

Reject Transit Gateway Vpc Attachment

Usage

ec2_reject_transit_gateway_vpc_attachment(
    TransitGatewayAttachmentId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

TransitGatewayAttachmentId
Character. The ID of the attachment.

DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]

others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  Character. The region of the AWS service.

Value
A list object or a character vector

TransitGatewayAttachmentId
The ID of the attachment.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_reject_vpc_endpoint_connections
Reject Vpc Endpoint Connections

Description
Rejects one or more VPC endpoint connection requests to your VPC endpoint service.

Usage
```
ec2_reject_vpc_endpoint_connections(
    ServiceId,
    VpcEndpointId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)```
Arguments

ServiceId  Character. The ID of the service.
VpcEndpointId  List. The IDs of one or more VPC endpoints.
DryRun  Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
simplify  Logical. Whether to simplify the result and handle nextToken in the response [optional]
others  Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]
print_on_error  Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region  Character. The region of the AWS service.

Value

A list object or a character vector

ServiceId

The ID of the service.

VpcEndpointId

The IDs of one or more VPC endpoints.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description

Rejects a VPC peering connection request. The VPC peering connection must be in the pending-acceptance state. Use the DescribeVpcPeeringConnections request to view your outstanding VPC peering connection requests. To delete an active VPC peering connection, or to delete a VPC peering connection request that you initiated, use DeleteVpcPeeringConnection.
Usage

```r
e2_reject_vpc_peering_connection(
  VpcPeeringConnectionId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **VpcPeeringConnectionId**
  Character. The ID of the VPC peering connection.
- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]
- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]
- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.
- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**
  Character. The region of the AWS service.

Value

A list object or a character vector

**VpcPeeringConnectionId**

The ID of the VPC peering connection.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`. 
ec2_release_address

**Description**

Release Address

**Usage**

```r
e2_release_address(  
  AllocationId = NULL,  
  PublicIp = NULL,  
  NetworkBorderGroup = NULL,  
  DryRun = NULL,  
  simplify = TRUE,  
  others = list(),  
  print_on_error = aws_get_print_on_error(),  
  retry_time = aws_get_retry_time(),  
  network_timeout = aws_get_network_timeout(),  
  region = aws_get_region()  
)
```

**Arguments**

- **AllocationId** Character. [EC2-VPC] The allocation ID. Required for EC2-VPC. [optional]
- **PublicIp** Character. [EC2-Classic] The Elastic IP address. Required for EC2-Classic. [optional]
- **NetworkBorderGroup** Character. The set of Availability Zones, Local Zones, or Wavelength Zones from which AWS advertises IP addresses. [optional]
- **DryRun** Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **simplify** Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]
- **others** Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]
- **print_on_error** Logical. Whether to show an error message when a network error occurs.
- **retry_time** Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout** Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region** Character. The region of the AWS service.

**Value**

A list object or a character vector
AllocationId

[EC2-VPC] The allocation ID. Required for EC2-VPC.

PublicIp


NetworkBorderGroup

The set of Availability Zones, Local Zones, or Wavelength Zones from which AWS advertises IP addresses.

If you provide an incorrect network border group, you will receive an InvalidAddress.NotFound error. For more information, see Error Codes.

You cannot use a network border group with EC2 Classic. If you attempt this operation on EC2 classic, you will receive an InvalidParameterCombination error. For more information, see Error Codes.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

ec2_release_hosts Release Hosts

Description

Release Hosts

Usage

```r
ec2_release_hosts(
    HostId,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```
Arguments

- **HostId**
  List. The IDs of the Dedicated Hosts to release.

- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response (optional).

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request (optional).

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounters the network issue. If the request has been sent `retry_time` times but still not able to get the response, an error will be thrown.

- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.

- **region**
  Character. The region of the AWS service.

Value

A list object or a character vector

**HostId**

The IDs of the Dedicated Hosts to release.

---

**ec2_replace_iam_instance_profile_association**

*Replace Iam Instance Profile Association*

**Description**

Replace Iam Instance Profile Association

**Usage**

```r
ec2_replace_iam_instance_profile_association(
  IamInstanceProfile,
  AssociationId,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
Arguments

IamInstanceProfile
Object. The IAM instance profile.

AssociationId
Character. The ID of the existing IAM instance profile association.

simplify
Logical. Whether to simplify the result and handle `nextToken` in the response [optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value
A list object or a character vector

IamInstanceProfile
The IAM instance profile.

AssociationId
The ID of the existing IAM instance profile association.

ec2_replace_network_acl_association
Replace Network Acl Association

Usage

```r
ec2_replace_network_acl_association(
  AssociationId,
  NetworkAclId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
)```
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

AssociationId  Character. The ID of the current association between the original network ACL and the subnet.

NetworkAclId   Character. The ID of the new network ACL to associate with the subnet.

DryRun        Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

simplify      Logical. Whether to simplify the result and handle nextToken in the response[optional]

others        Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time    Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region        Character. The region of the AWS service.

Value

A list object or a character vector

AssociationId

The ID of the current association between the original network ACL and the subnet.

NetworkAclId

The ID of the new network ACL to associate with the subnet.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
ec2_replace_network_acl_entry

Replace Network Acl Entry

Description

Replaces an entry (rule) in a network ACL. For more information, see Network ACLs in the Amazon Virtual Private Cloud User Guide.

Usage

ec2_replace_network_acl_entry(
    Egress,
    NetworkAclId,
    Protocol,
    RuleAction,
    RuleNumber,
    CidrBlock = NULL,
    DryRun = NULL,
    Icmp = NULL,
    Ipv6CidrBlock = NULL,
    PortRange = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

Egress Logical. Indicates whether to replace the egress rule.
NetworkAclId Character. The ID of the ACL.
Protocol Character. The protocol number.
RuleAction Character. Indicates whether to allow or deny the traffic that matches the rule.
RuleNumber Integer. The rule number of the entry to replace.
CidrBlock Character. The IPv4 network range to allow or deny, in CIDR notation (for example 172.16.0.0/24).[optional]
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]
Icmp Object. ICMP protocol: The ICMP or ICMPv6 type and code.[optional]
Ipv6CidrBlock Character. The IPv6 network range to allow or deny, in CIDR notation (for example 2001:bd8:1234:1a00::/64).[optional]
PortRange  Object. TCP or UDP protocols: The range of ports the rule applies to.[optional]
simplify  Logical. Whether to simplify the result and handle nextToken in the response[optional]
others  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error  Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region  Character. The region of the AWS service.

Value
A list object or a character vector

Egress
Indicates whether to replace the egress rule.
Default: If no value is specified, we replace the ingress rule.

NetworkAclId
The ID of the ACL.

Protocol
The protocol number. A value of `-1` means all protocols. If you specify `-1` or a protocol number other than `6` (TCP), `17` (UDP), or `1` (ICMP), traffic on all ports is allowed, regardless of any ports or ICMP types or codes that you specify. If you specify protocol `58` (ICMPv6) and specify an IPv4 CIDR block, traffic for all ICMP types and codes allowed, regardless of any that you specify. If you specify protocol `58` (ICMPv6) and specify an IPv6 CIDR block, you must specify an ICMP type and code.

RuleAction
Indicates whether to allow or deny the traffic that matches the rule.

RuleNumber
The rule number of the entry to replace.

CidrBlock
The IPv4 network range to allow or deny, in CIDR notation (for example 172.16.0.0/24).
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Icmp

ICMP protocol: The ICMP or ICMPv6 type and code. Required if specifying protocol 1 (ICMP) or protocol 58 (ICMPv6) with an IPv6 CIDR block.

Ipv6CidrBlock

The IPv6 network range to allow or deny, in CIDR notation (for example 2001:bd8:1234:1a00::/64).

PortRange

TCP or UDP protocols: The range of ports the rule applies to. Required if specifying protocol 6 (TCP) or 17 (UDP).

ec2_replace_route

Replace Route

Description

Replace Route

Usage

ec2_replace_route(
    RouteTableId,
    DestinationCidrBlock = NULL,
    DestinationIpv6CidrBlock = NULL,
    DestinationPrefixListId = NULL,
    DryRun = NULL,
    VpcEndpointId = NULL,
    EgressOnlyInternetGatewayId = NULL,
    GatewayId = NULL,
    InstanceId = NULL,
    LocalTarget = NULL,
    NatGatewayId = NULL,
    TransitGatewayId = NULL,
    LocalGatewayId = NULL,
    CarrierGatewayId = NULL,
    NetworkInterfaceId = NULL,
    VpcPeeringConnectionId = NULL,
    simplify = TRUE,
    others = list(),
)
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

RouteTableId Character. The ID of the route table.
DestinationCidrBlock
   Character. The IPv4 CIDR address block used for the destination match.[optional]
DestinationIpv6CidrBlock
   Character. The IPv6 CIDR address block used for the destination match.[optional]
DestinationPrefixListId
   Character. The ID of the prefix list for the route.[optional]
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]
VpcEndpointId Character. The ID of a VPC endpoint. Supported for Gateway Load Balancer endpoints only.[optional]
EgressOnlyInternetGatewayId
   Character. [IPv6 traffic only] The ID of an egress-only internet gateway.[optional]
GatewayId Character. The ID of an internet gateway or virtual private gateway.[optional]
InstanceId Character. The ID of a NAT instance in your VPC.[optional]
LocalTarget Logical. Specifies whether to reset the local route to its default target (local).[optional]
NatGatewayId Character. [IPv4 traffic only] The ID of a NAT gateway.[optional]
TransitGatewayId
   Character. The ID of a transit gateway.[optional]
LocalGatewayId Character. The ID of the local gateway.[optional]
CarrierGatewayId Character. [IPv4 traffic only] The ID of a carrier gateway.[optional]
NetworkInterfaceId
   Character. The ID of a network interface.[optional]
VpcPeeringConnectionId
   Character. The ID of a VPC peering connection.[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.
Value
A list object or a character vector

RouteTableId
The ID of the route table.

DestinationCidrBlock
The IPv4 CIDR address block used for the destination match. The value that you provide must match the CIDR of an existing route in the table.

DestinationIpv6CidrBlock
The IPv6 CIDR address block used for the destination match. The value that you provide must match the CIDR of an existing route in the table.

DestinationPrefixListId
The ID of the prefix list for the route.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

VpcEndpointId
The ID of a VPC endpoint. Supported for Gateway Load Balancer endpoints only.

EgressOnlyInternetGatewayId
[IPv6 traffic only] The ID of an egress-only internet gateway.

GatewayId
The ID of an internet gateway or virtual private gateway.

InstanceId
The ID of a NAT instance in your VPC.

LocalTarget
Specifies whether to reset the local route to its default target (local).

NatGatewayId
[IPv4 traffic only] The ID of a NAT gateway.
TransitGatewayId

The ID of a transit gateway.

LocalGatewayId

The ID of the local gateway.

CarrierGatewayId

[IPv4 traffic only] The ID of a carrier gateway.

NetworkInterfaceId

The ID of a network interface.

VpcPeeringConnectionId

The ID of a VPC peering connection.

---

**Description**

Replace Route Table Association

**Usage**

```r
ec2_replace_route_table_association(
  AssociationId,
  RouteTableId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
ec2_replace_transit_gateway_route

Replace Transit Gateway Route

Description

Replaces the specified route in the specified transit gateway route table.
Usage

```r
e2_replace_transit_gateway_route(
  DestinationCidrBlock, 
  TransitGatewayRouteTableId, 
  TransitGatewayAttachmentId = NULL, 
  Blackhole = NULL, 
  DryRun = NULL, 
  simplify = TRUE, 
  others = list(), 
  print_on_error = aws_get_print_on_error(), 
  retry_time = aws_get_retry_time(), 
  network_timeout = aws_get_network_timeout(), 
  region = aws_get_region()
)
```

Arguments

- **DestinationCidrBlock**: Character. The CIDR range used for the destination match.
- **TransitGatewayRouteTableId**: Character. The ID of the route table.
- **TransitGatewayAttachmentId**: Character. The ID of the attachment. [optional]
- **Blackhole**: Logical. Indicates whether traffic matching this route is to be dropped. [optional]
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encountering the network issue. If the request has been sent `retry_time` times but still not able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

**DestinationCidrBlock**

The CIDR range used for the destination match. Routing decisions are based on the most specific match.
**TransitGatewayRouteTableId**

The ID of the route table.

**TransitGatewayAttachmentId**

The ID of the attachment.

**Blackhole**

Indicates whether traffic matching this route is to be dropped.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

---

```r
ec2_report_instance_status

Report Instance Status

Description

Report Instance Status

Usage

ec2_report_instance_status(
  InstanceId,
  ReasonCode,
  Status,
  Description = NULL,
  DryRun = NULL,
  EndTime = NULL,
  StartTime = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
Arguments

InstanceId List. The instances.
ReasonCode List. The reason codes that describe the health state of your instance.
Status Character. The status of all instances listed.
Description Character. Descriptive text about the health state of your instance. [optional]
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
EndTime Character. The time at which the reported instance health state ended. [optional]
StartTime Character. The time at which the reported instance health state began. [optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response. [optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value

A list object or a character vector

InstanceId

The instances.

ReasonCode

The reason codes that describe the health state of your instance.

- instance-stuck-in-state: My instance is stuck in a state.
- unresponsive: My instance is unresponsive.
- not-accepting-credentials: My instance is not accepting my credentials.
- password-not-available: A password is not available for my instance.
- performance-network: My instance is experiencing performance problems that I believe are network related.
- performance-instance-store: My instance is experiencing performance problems that I believe are related to the instance stores.
- performance-ebs-volume: My instance is experiencing performance problems that I believe are related to an EBS volume.
- performance-other: My instance is experiencing performance problems.
- other: [explain using the description parameter]
Status
The status of all instances listed.

Description
Descriptive text about the health state of your instance.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

EndTime
The time at which the reported instance health state ended.

StartTime
The time at which the reported instance health state began.

---

```
ec2_request_spot_fleet

Request Spot Fleet
```

Description
Request Spot Fleet

Usage
```
ec2_request_spot_fleet(
    SpotFleetRequestConfig,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```
ec2_request_spot_instances

Arguments

SpotFleetRequestConfig
Object. The configuration for the Spot Fleet request.

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value

A list object or a character vector

SpotFleetRequestConfig

The configuration for the Spot Fleet request.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description

Request Spot Instances
Usage

```r
ec2_request_spot_instances(
    AvailabilityZoneGroup = NULL,
    BlockDurationMinutes = NULL,
    ClientToken = NULL,
    DryRun = NULL,
    InstanceCount = NULL,
    LaunchGroup = NULL,
    LaunchSpecification = NULL,
    SpotPrice = NULL,
    Type = NULL,
    ValidFrom = NULL,
    ValidUntil = NULL,
    TagSpecification = NULL,
    InstanceInterruptionBehavior = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- **AvailabilityZoneGroup**
  - Character. The user-specified name for a logical grouping of requests.[optional]

- **BlockDurationMinutes**
  - Integer. The required duration for the Spot Instances (also known as Spot blocks), in minutes.[optional]

- **ClientToken**
  - Character. Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.[optional]

- **DryRun**
  - Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

- **InstanceCount**
  - Integer. The maximum number of Spot Instances to launch. Default: 1 [optional]

- **LaunchGroup**
  - Character. The instance launch group.[optional]

- **LaunchSpecification**
  - Object. The launch specification.[optional]

- **SpotPrice**
  - Character. The maximum price per hour that you are willing to pay for a Spot Instance.[optional]

- **Type**
  - Character. The Spot Instance request type. Default: one-time [optional]

- **ValidFrom**
  - Character. The start date of the request.[optional]

- **ValidUntil**
  - Character. The end date of the request, in UTC format (YYYY-MM-DDTHH:MM:SSZ).[optional]

- **TagSpecification**
  - List. The key-value pair for tagging the Spot Instance request on creation.[optional]
InstanceInterruptionBehavior

Character. The behavior when a Spot Instance is interrupted. The default is terminate.[optional]

simplify

Logical. Whether to simplify the result and handle nextToken in the response[optional]

others

Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error

Logical. Whether to show an error message when a network error occurs.

retry_time

Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout

Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region

Character. The region of the AWS service.

Value

A list object or a character vector

AvailabilityZoneGroup

The user-specified name for a logical grouping of requests.

When you specify an Availability Zone group in a Spot Instance request, all Spot Instances in the request are launched in the same Availability Zone. Instance proximity is maintained with this parameter, but the choice of Availability Zone is not. The group applies only to requests for Spot Instances of the same instance type. Any additional Spot Instance requests that are specified with the same Availability Zone group name are launched in that same Availability Zone, as long as at least one instance from the group is still active.

If there is no active instance running in the Availability Zone group that you specify for a new Spot Instance request (all instances are terminated, the request is expired, or the maximum price you specified falls below current Spot price), then Amazon EC2 launches the instance in any Availability Zone where the constraint can be met. Consequently, the subsequent set of Spot Instances could be placed in a different zone from the original request, even if you specified the same Availability Zone group.

Default: Instances are launched in any available Availability Zone.

BlockDurationMinutes

The required duration for the Spot Instances (also known as Spot blocks), in minutes. This value must be a multiple of 60 (60, 120, 180, 240, 300, or 360).

The duration period starts as soon as your Spot Instance receives its instance ID. At the end of the duration period, Amazon EC2 marks the Spot Instance for termination and provides a Spot Instance termination notice, which gives the instance a two-minute warning before it terminates.

You can\'t specify an Availability Zone group or a launch group if you specify a duration.

New accounts or accounts with no previous billing history with AWS are not eligible for Spot Instances with a defined duration (also known as Spot blocks).
ClientToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. For more information, see How to Ensure Idempotency in the Amazon EC2 User Guide for Linux Instances.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

InstanceCount

The maximum number of Spot Instances to launch.

Default: 1

LaunchGroup

The instance launch group. Launch groups are Spot Instances that launch together and terminate together.

Default: Instances are launched and terminated individually

LaunchSpecification

The launch specification.

SpotPrice

The maximum price per hour that you are willing to pay for a Spot Instance. The default is the On-Demand price.

Type

The Spot Instance request type.

Default: one-time

ValidFrom

The start date of the request. If this is a one-time request, the request becomes active at this date and time and remains active until all instances launch, the request expires, or the request is canceled. If the request is persistent, the request becomes active at this date and time and remains active until it expires or is canceled.

The specified start date and time cannot be equal to the current date and time. You must specify a start date and time that occurs after the current date and time.
ValidUntil

The end date of the request, in UTC format (YYYY-MM-DDTHH:MM:SSZ).

- For a persistent request, the request remains active until the ValidUntil date and time is reached. Otherwise, the request remains active until you cancel it.
- For a one-time request, the request remains active until all instances launch, the request is canceled, or the ValidUntil date and time is reached. By default, the request is valid for 7 days from the date the request was created.

TagSpecification

The key-value pair for tagging the Spot Instance request on creation. The value for ResourceType must be spot-instances-request, otherwise the Spot Instance request fails. To tag the Spot Instance request after it has been created, see CreateTags.

InstanceInterruptionBehavior

The behavior when a Spot Instance is interrupted. The default is terminate.

---

description

Resets the attribute of the specified IP address. For requirements, see Using reverse DNS for email applications.

Usage

```r
ec2_reset_address_attribute(
  AllocationId,
  Attribute,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
Arguments

AllocationId  Character. [EC2-VPC] The allocation ID.
Attribute     Character. The attribute of the IP address.
DryRun        Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
simplify     Logical. Whether to simplify the result and handle nextToken in the response[optional]
others        Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time    Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region          Character. The region of the AWS service.

Value

A list object or a character vector

AllocationId

[EC2-VPC] The allocation ID.

Attribute

The attribute of the IP address.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description

Reset Ebs Default Kms Key Id
Usage

```r
ec2_reset_ebs_default_kms_key_id(  
  DryRun = NULL,  
  simplify = TRUE,  
  others = list(),  
  print_on_error = aws_get_print_on_error(),  
  retry_time = aws_get_retry_time(),  
  network_timeout = aws_get_network_timeout(),  
  region = aws_get_region()
)
```

Arguments

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]
- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]
- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.
- **retry_time**
  Integer. Number of retries for a REST request when encountering the network issue. If the request has been sent `retry_time` times but still not able to get the response, an error will be thrown.
- **network_timeout**
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**
  Character. The region of the AWS service.

Value

A list object or a character vector

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

Description

Resets the specified attribute of the specified Amazon FPGA Image (AFI) to its default value. You can only reset the load permission attribute.
Usage

```r
e2_reset_fpga_image_attribute(
  FpgaImageId,
  DryRun = NULL,
  Attribute = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **FpgaImageId**: Character. The ID of the AFI.
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **Attribute**: Character. The attribute. [optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

**FpgaImageId**

The ID of the AFI.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is `DryRunOperation`. Otherwise, it is `UnauthorizedOperation`.

**Attribute**

The attribute.
ec2_reset_image_attribute

Reset Image Attribute

Description

Reset Image Attribute

Usage

```r
ec2_reset_image_attribute(
  Attribute,
  ImageId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **Attribute**: Character. The attribute to reset (currently you can only reset the launch permission attribute).
- **ImageId**: Character. The ID of the AMI.
- **DryRun**: Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector
Attribute

The attribute to reset (currently you can only reset the launch permission attribute).

ImageId

The ID of the AMI.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description

Reset Instance Attribute

Usage

```r
ec2_reset_instance_attribute(
    Attribute,
    InstanceId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attribute</td>
<td>Character. The attribute to reset.</td>
</tr>
<tr>
<td>InstanceId</td>
<td>Character. The ID of the instance.</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request,[optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response[optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
</tbody>
</table>
**retry_time**
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

**network_timeout**
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

**region**
Character. The region of the AWS service.

**Value**
A list object or a character vector

**Attribute**
The attribute to reset.
You can only reset the following attributes: kernel | ramdisk | sourceDestCheck. To change an instance attribute, use ModifyInstanceAttribute.

**InstanceId**
The ID of the instance.

**DryRun**
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**ec2_reset_network_interface_attribute**

*Reset Network Interface Attribute*

---

**Description**
Resets a network interface attribute. You can specify only one attribute at a time.

**Usage**

```r
ec2_reset_network_interface_attribute(
    NetworkInterfaceId,
    DryRun = NULL,
    SourceDestCheck = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```
Arguments

**NetworkInterfaceId**
- Character. The ID of the network interface.

**DryRun**
- Logical. Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is **DryRunOperation**. Otherwise, it is **UnauthorizedOperation**.

**SourceDestCheck**
- Character. The source/destination checking attribute. Resets the value to true.

**simplify**
- Logical. Whether to simplify the result and handle `nextToken` in the response.

**others**
- Named list. The parameters that are not included in the function parameters and need to be added into the request.

**print_on_error**
- Logical. Whether to show an error message when a network error occurs.

**retry_time**
- Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

**network_timeout**
- Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.

**region**
- Character. The region of the AWS service.

Value

- A list object or a character vector

**NetworkInterfaceId**

The ID of the network interface.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is **DryRunOperation**. Otherwise, it is **UnauthorizedOperation**.

**SourceDestCheck**

The source/destination checking attribute. Resets the value to true.
Description

Reset Snapshot Attribute

Usage

```r
ec2_reset_snapshot_attribute(
  Attribute,
  SnapshotId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attribute</td>
<td>Character. The attribute to reset.</td>
</tr>
<tr>
<td>SnapshotId</td>
<td>Character. The ID of the snapshot.</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response. [optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>

Value

A list object or a character vector
**Attribute**

The attribute to reset. Currently, only the attribute for permission to create volumes can be reset.

**SnapshotId**

The ID of the snapshot.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**Usage**

```r
ec2_restore_address_to_classic(
  PublicIp,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- **PublicIp**  Character. The Elastic IP address.
- **DryRun** Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
- **simplify** Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others** Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error** Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  Character. The region of the AWS service.

Value
A list object or a character vector

PublicIp
The Elastic IP address.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

ec2_restore_managed_prefix_list_version

*Restore Managed Prefix List Version*

**Description**
Restores the entries from a previous version of a managed prefix list to a new version of the prefix list.

**Usage**
```r
ec2_restore_managed_prefix_list_version(
    PrefixListId,
    PreviousVersion,
    CurrentVersion,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```
Arguments

PrefixListId  Character. The ID of the prefix list.
PreviousVersion  Integer. The version to restore.
CurrentVersion  Integer. The current version number for the prefix list.
DryRun  Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
simplify  Logical. Whether to simplify the result and handle nextToken in the response[optional]
others  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error  Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region  Character. The region of the AWS service.

Value

A list object or a character vector

PrefixListId

The ID of the prefix list.

PreviousVersion

The version to restore.

CurrentVersion

The current version number for the prefix list.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
Revoke Client Vpn Ingress

Description

Removes an ingress authorization rule from a Client VPN endpoint.

Usage

```r
ec2_revoke_client_vpn_ingress(
  ClientVpnEndpointId,  
  TargetNetworkCidr, 
  AccessGroupId = NULL, 
  RevokeAllGroups = NULL, 
  DryRun = NULL, 
  simplify = TRUE, 
  others = list(), 
  print_on_error = aws_get_print_on_error(), 
  retry_time = aws_get_retry_time(), 
  network_timeout = aws_get_network_timeout(), 
  region = aws_get_region()
)
```

Arguments

- **ClientVpnEndpointId**
  Character. The ID of the Client VPN endpoint with which the authorization rule is associated.

- **TargetNetworkCidr**
  Character. The IPv4 address range, in CIDR notation, of the network for which access is being removed.

- **AccessGroupId**
  Character. The ID of the Active Directory group for which to revoke access. [optional]

- **RevokeAllGroups**
  Logical. Indicates whether access should be revoked for all clients. [optional]

- **DryRun**
  Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]

- **simplify**
  Logical. Whether to simplify the result and handle `nextToken` in the response. [optional]

- **others**
  Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]

- **print_on_error**
  Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
network_timeout
   Numeric. Number of seconds to wait for a REST response until giving up. Can
   not be less than 1 ms.

region
   Character. The region of the AWS service.

Value
   A list object or a character vector

ClientVpnEndpointId
   The ID of the Client VPN endpoint with which the authorization rule is associated.

TargetNetworkCidr
   The IPv4 address range, in CIDR notation, of the network for which access is being removed.

AccessGroupId
   The ID of the Active Directory group for which to revoke access.

RevokeAllGroups
   Indicates whether access should be revoked for all clients.

DryRun
   Checks whether you have the required permissions for the action, without actually making the
   request, and provides an error response. If you have the required permissions, the error response is
   DryRunOperation. Otherwise, it is UnauthorizedOperation.

ec2_revoke_security_group_egress
   Revoke Security Group Egress

Description
   Revoke Security Group Egress

Usage
   ec2_revoke_security_group_egress(
       GroupId,
       DryRun = NULL,
       IpPermissions = NULL,
       CidrIp = NULL,
       FromPort = NULL,
       IpProtocol = NULL,
ToPort = NULL,
SourceSecurityGroupName = NULL,
SourceSecurityGroupOwnerId = NULL,
simplify = TRUE,
others = list(),
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

GroupId Character. The ID of the security group.
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
IpPermissions List. The sets of IP permissions.[optional]
CidrIp Character. Not supported. Use a set of IP permissions to specify the CIDR.[optional]
FromPort Integer. Not supported. Use a set of IP permissions to specify the port.[optional]
IpProtocol Character. Not supported. Use a set of IP permissions to specify the protocol name or number.[optional]
ToPort Integer. Not supported. Use a set of IP permissions to specify the port.[optional]
SourceSecurityGroupName Character. Not supported. Use a set of IP permissions to specify a destination security group.[optional]
SourceSecurityGroupOwnerId Character. Not supported. Use a set of IP permissions to specify a destination security group.[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value

A list object or a character vector
**GroupId**

The ID of the security group.

**DryRun**

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

**IpPermissions**

The sets of IP permissions. You can't specify a destination security group and a CIDR IP address range in the same set of permissions.

**CidrIp**

Not supported. Use a set of IP permissions to specify the CIDR.

**FromPort**

Not supported. Use a set of IP permissions to specify the port.

**IpProtocol**

Not supported. Use a set of IP permissions to specify the protocol name or number.

**ToPort**

Not supported. Use a set of IP permissions to specify the port.

**SourceSecurityGroupName**

Not supported. Use a set of IP permissions to specify a destination security group.

**SourceSecurityGroupOwnerId**

Not supported. Use a set of IP permissions to specify a destination security group.
ec2_revoke_security_group_ingress

Revoke Security Group Ingress

Description

Revoke Security Group Ingress

Usage

ec2_revoke_security_group_ingress(
    CidrIp = NULL,
    FromPort = NULL,
    GroupId = NULL,
    GroupName = NULL,
    IpPermissions = NULL,
    IpProtocol = NULL,
    SourceSecurityGroupName = NULL,
    SourceSecurityGroupOwnerId = NULL,
    ToPort = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

CidrIp          Character. The CIDR IP address range.[optional]
FromPort       Integer. The start of port range for the TCP and UDP protocols, or an ICMP
type number.[optional]
GroupId        Character. The ID of the security group.[optional]
GroupName      Character. [EC2-Classic, default VPC] The name of the security group.[optional]
IpPermissions  List. The sets of IP permissions.[optional]
IpProtocol     Character. The IP protocol name (tcp, udp, icmp) or number (see Protocol
Numbers)...[optional]
SourceSecurityGroupName
Character. [EC2-Classic, default VPC] The name of the source security group.[optional]
SourceSecurityGroupOwnerId
Character. [EC2-Classic] The AWS account ID of the source security group, if the source security group is...[optional]
ToPort  
Integer. The end of port range for the TCP and UDP protocols, or an ICMP code number.[optional]

DryRun  
Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

simplify  
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others  
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error  
Logical. Whether to show an error message when a network error occurs.

retry_time  
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  
Character. The region of the AWS service.

Value
A list object or a character vector

CidrIp
The CIDR IP address range. You can\'t specify this parameter when specifying a source security group.

FromPort
The start of port range for the TCP and UDP protocols, or an ICMP type number. For the ICMP type number, use -1 to specify all ICMP types.

GroupId
The ID of the security group. You must specify either the security group ID or the security group name in the request. For security groups in a nondefault VPC, you must specify the security group ID.

GroupName
[EC2-Classic, default VPC] The name of the security group. You must specify either the security group ID or the security group name in the request.

IpPermissions
The sets of IP permissions. You can\'t specify a source security group and a CIDR IP address range in the same set of permissions.

IpProtocol
The IP protocol name (tcp, udp, icmp) or number (see Protocol Numbers). Use -1 to specify all.
SourceSecurityGroupName

[EC2-Classic, default VPC] The name of the source security group. You can’t specify this parameter in combination with the following parameters: the CIDR IP address range, the start of the port range, the IP protocol, and the end of the port range. For EC2-VPC, the source security group must be in the same VPC. To revoke a specific rule for an IP protocol and port range, use a set of IP permissions instead.

SourceSecurityGroupOwnerId

[EC2-Classic] The AWS account ID of the source security group, if the source security group is in a different account. You can’t specify this parameter in combination with the following parameters: the CIDR IP address range, the IP protocol, the start of the port range, and the end of the port range. To revoke a specific rule for an IP protocol and port range, use a set of IP permissions instead.

ToPort

The end of port range for the TCP and UDP protocols, or an ICMP code number. For the ICMP code number, use -1 to specify all ICMP codes for the ICMP type.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

**ec2_run_instances**

**Run Instances**

**Description**

Run Instances

**Usage**

```python
ec2_run_instances(
    MaxCount,
    MinCount,
    BlockDeviceMapping = NULL,
    ImageId = NULL,
    InstanceType = NULL,
    Ipv6AddressCount = NULL,
    Ipv6Address = NULL,
    KernelId = NULL,
    KeyName = NULL,
    Monitoring = NULL,
    Placement = NULL,
    RamdiskId = NULL,
```

SecurityGroupId = NULL,
SecurityGroup = NULL,
SubnetId = NULL,
UserData = NULL,
AdditionalInfo = NULL,
ClientToken = NULL,
DisableApiTermination = NULL,
DryRun = NULL,
EbsOptimized = NULL,
IamInstanceProfile = NULL,
InstanceInitiatedShutdownBehavior = NULL,
NetworkInterface = NULL,
PrivateIpAddress = NULL,
ElasticGpuSpecification = NULL,
ElasticInferenceAccelerator = NULL,
TagSpecification = NULL,
LaunchTemplate = NULL,
InstanceMarketOptions = NULL,
CreditSpecification = NULL,
CpuOptions = NULL,
CapacityReservationSpecification = NULL,
HibernationOptions = NULL,
LicenseSpecification = NULL,
MetadataOptions = NULL,
EnclaveOptions = NULL,
simplify = TRUE,
others = list(),
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

MaxCount  Integer. The maximum number of instances to launch.
MinCount  Integer. The minimum number of instances to launch.
BlockDeviceMapping
  List. The block device mapping entries.[optional]
ImageId    Character. The ID of the AMI.[optional]
InstanceType  Character. The instance type.[optional]
Ipv6AddressCount  Integer. [EC2-VPC] The number of IPv6 addresses to associate with the primary network interface.[optional]
Ipv6Address        List. [EC2-VPC] The IPv6 addresses from the range of the subnet to associate with the primary network...[optional]
KernelId           Character. The ID of the kernel.[optional]
KeyName
Character. The name of the key pair.[optional]
Monitoring
Object. Specifies whether detailed monitoring is enabled for the instance.[optional]
Placement
Object. The placement for the instance.[optional]
RamdiskId
Character. The ID of the RAM disk to select.[optional]
SecurityGroupId
List. The IDs of the security groups.[optional]
SecurityGroup
List. [EC2-Classic, default VPC] The names of the security groups.[optional]
SubnetId
Character. [EC2-VPC] The ID of the subnet to launch the instance into.[optional]
UserData
Character. The user data to make available to the instance.[optional]
AdditionalInfo
Character. Reserved.[optional]
ClientToken
Character. Unique, case-sensitive identifier you provide to ensure the idempotency of the request.[optional]
DisableApiTermination
Logical. If you set this parameter to true, you can’t terminate the instance using the Amazon EC2 console...[optional]
DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]
EbsOptimized
Logical. Indicates whether the instance is optimized for Amazon EBS I/O.[optional]
IamInstanceProfile
Object. The name or Amazon Resource Name (ARN) of an IAM instance profile.[optional]
InstanceInitiatedShutdownBehavior
Character. Indicates whether an instance stops or terminates when you initiate shutdown from the instance...[optional]
NetworkInterface
List. The network interfaces to associate with the instance.[optional]
PrivateIpAddress
Character. [EC2-VPC] The primary IPv4 address.[optional]
ElasticGpuSpecification
List. An elastic GPU to associate with the instance.[optional]
ElasticInferenceAccelerator
List. An elastic inference accelerator to associate with the instance.[optional]
TagSpecification
List. The tags to apply to the resources during launch.[optional]
LaunchTemplate
Object. The launch template to use to launch the instances.[optional]
InstanceMarketOptions
Object. The market (purchasing) option for the instances.[optional]
CreditSpecification
Object. The credit option for CPU usage of the burstable performance instance.[optional]
CpuOptions
Object. The CPU options for the instance.[optional]
CapacityReservationSpecification
Object. Information about the Capacity Reservation targeting option.[optional]
HibernationOptions
Object. Indicates whether an instance is enabled for hibernation.[optional]

LicenseSpecification
List. The license configurations.[optional]

MetadataOptions
Object. The metadata options for the instance.[optional]

EnclaveOptions
Object. Indicates whether the instance is enabled for AWS Nitro Enclaves.[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value
A list object or a character vector

MaxCount
The maximum number of instances to launch. If you specify more instances than Amazon EC2 can launch in the target Availability Zone, Amazon EC2 launches the largest possible number of instances above MinCount.

Constraints: Between 1 and the maximum number you’re allowed for the specified instance type. For more information about the default limits, and how to request an increase, see How many instances can I run in Amazon EC2 in the Amazon EC2 FAQ.

MinCount
The minimum number of instances to launch. If you specify a minimum that is more instances than Amazon EC2 can launch in the target Availability Zone, Amazon EC2 launches no instances.

Constraints: Between 1 and the maximum number you’re allowed for the specified instance type. For more information about the default limits, and how to request an increase, see How many instances can I run in Amazon EC2 in the Amazon EC2 General FAQ.

BlockDeviceMapping
The block device mapping entries.

ImageId
The ID of the AMI. An AMI ID is required to launch an instance and must be specified here or in a launch template.
**InstanceType**

The instance type. For more information, see Instance types in the Amazon EC2 User Guide.

Default: m1.small

**Ipv6AddressCount**

[EC2-VPC] The number of IPv6 addresses to associate with the primary network interface. Amazon EC2 chooses the IPv6 addresses from the range of your subnet. You cannot specify this option and the option to assign specific IPv6 addresses in the same request. You can specify this option if you’ve specified a minimum number of instances to launch.

You cannot specify this option and the network interfaces option in the same request.

**Ipv6Address**

[EC2-VPC] The IPv6 addresses from the range of the subnet to associate with the primary network interface. You cannot specify this option and the option to assign a number of IPv6 addresses in the same request. You cannot specify this option if you’ve specified a minimum number of instances to launch.

You cannot specify this option and the network interfaces option in the same request.

**KernelId**

The ID of the kernel.

We recommend that you use PV-GRUB instead of kernels and RAM disks. For more information, see PV-GRUB in the Amazon EC2 User Guide.

**KeyName**

The name of the key pair. You can create a key pair using CreateKeyPair or ImportKeyPair.

If you do not specify a key pair, you can’t connect to the instance unless you choose an AMI that is configured to allow users another way to log in.

**Monitoring**

Specifies whether detailed monitoring is enabled for the instance.

**Placement**

The placement for the instance.

**RamdiskId**

The ID of the RAM disk to select. Some kernels require additional drivers at launch. Check the kernel requirements for information about whether you need to specify a RAM disk. To find kernel requirements, go to the AWS Resource Center and search for the kernel ID.

We recommend that you use PV-GRUB instead of kernels and RAM disks. For more information, see PV-GRUB in the Amazon EC2 User Guide.
SecurityGroupId

The IDs of the security groups. You can create a security group using CreateSecurityGroup.

If you specify a network interface, you must specify any security groups as part of the network interface.

SecurityGroup

[EC2-Classic, default VPC] The names of the security groups. For a nondefault VPC, you must use security group IDs instead.

If you specify a network interface, you must specify any security groups as part of the network interface.

Default: Amazon EC2 uses the default security group.

SubnetId

[EC2-VPC] The ID of the subnet to launch the instance into.

If you specify a network interface, you must specify any subnets as part of the network interface.

UserData

The user data to make available to the instance. For more information, see Running commands on your Linux instance at launch (Linux) and Adding User Data (Windows). If you are using a command line tool, base64-encoding is performed for you, and you can load the text from a file. Otherwise, you must provide base64-encoded text. User data is limited to 16 KB.

AdditionalInfo

Reserved.

ClientToken

Unique, case-sensitive identifier you provide to ensure the idempotency of the request. If you do not specify a client token, a randomly generated token is used for the request to ensure idempotency.

For more information, see Ensuring Idempotency.

Constraints: Maximum 64 ASCII characters

DisableApiTermination

If you set this parameter to true, you can't terminate the instance using the Amazon EC2 console, CLI, or API; otherwise, you can. To change this attribute after launch, use ModifyInstanceAttribute. Alternatively, if you set InstanceInitiatedShutdownBehavior to terminate, you can terminate the instance by running the shutdown command from the instance.

Default: false
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

EbsOptimized

Indicates whether the instance is optimized for Amazon EBS I/O. This optimization provides dedicated throughput to Amazon EBS and an optimized configuration stack to provide optimal Amazon EBS I/O performance. This optimization isn't available with all instance types. Additional usage charges apply when using an EBS-optimized instance.

Default: false

IamInstanceProfile

The name or Amazon Resource Name (ARN) of an IAM instance profile.

InstanceInitiatedShutdownBehavior

Indicates whether an instance stops or terminates when you initiate shutdown from the instance (using the operating system command for system shutdown).

Default: stop

NetworkInterface

The network interfaces to associate with the instance. If you specify a network interface, you must specify any security groups and subnets as part of the network interface.

PrivateIpAddress

[EC2-VPC] The primary IPv4 address. You must specify a value from the IPv4 address range of the subnet.

Only one private IP address can be designated as primary. You can't specify this option if you've specified the option to designate a private IP address as the primary IP address in a network interface specification. You cannot specify this option if you're launching more than one instance in the request.

You cannot specify this option and the network interfaces option in the same request.

ElasticGpuSpecification

An elastic GPU to associate with the instance. An Elastic GPU is a GPU resource that you can attach to your Windows instance to accelerate the graphics performance of your applications. For more information, see Amazon EC2 Elastic GPUs in the Amazon EC2 User Guide.
ElasticInferenceAccelerator

An elastic inference accelerator to associate with the instance. Elastic inference accelerators are a resource you can attach to your Amazon EC2 instances to accelerate your Deep Learning (DL) inference workloads.

You cannot specify accelerators from different generations in the same request.

TagSpecification

The tags to apply to the resources during launch. You can only tag instances and volumes on launch. The specified tags are applied to all instances or volumes that are created during launch. To tag a resource after it has been created, see CreateTags.

LaunchTemplate

The launch template to use to launch the instances. Any parameters that you specify in RunInstances override the same parameters in the launch template. You can specify either the name or ID of a launch template, but not both.

InstanceMarketOptions

The market (purchasing) option for the instances.

For RunInstances, persistent Spot Instance requests are only supported when InstanceInterruptBehavior is set to either hibernate or stop.

CreditSpecification

The credit option for CPU usage of the burstable performance instance. Valid values are standard and unlimited. To change this attribute after launch, use ModifyInstanceCreditSpecification. For more information, see Burstable performance instances in the Amazon EC2 User Guide.

Default: standard (T2 instances) or unlimited (T3/T3a instances)

CpuOptions

The CPU options for the instance. For more information, see Optimizing CPU options in the Amazon EC2 User Guide.

CapacityReservationSpecification

Information about the Capacity Reservation targeting option. If you do not specify this parameter, the instance’s Capacity Reservation preference defaults to open, which enables it to run in any open Capacity Reservation that has matching attributes (instance type, platform, Availability Zone).

HibernationOptions

Indicates whether an instance is enabled for hibernation. For more information, see Hibernate your instance in the Amazon EC2 User Guide.

You can’t enable hibernation and AWS Nitro Enclaves on the same instance.
LicenseSpecification

The license configurations.

MetadataOptions

The metadata options for the instance. For more information, see Instance metadata and user data.

EnclaveOptions

Indicates whether the instance is enabled for AWS Nitro Enclaves. For more information, see What is AWS Nitro Enclaves? in the AWS Nitro Enclaves User Guide.

You can’t enable AWS Nitro Enclaves and hibernation on the same instance.

---

**Run Scheduled Instances**

**Description**

Run Scheduled Instances

**Usage**

```r
ec2_run_scheduled_instances(
    LaunchSpecification,
    ScheduledInstanceId,
    ClientToken = NULL,
    DryRun = NULL,
    InstanceCount = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

**Arguments**

- **LaunchSpecification**
  Object. The launch specification.

- **ScheduledInstanceId**
  Character. The Scheduled Instance ID.

- **ClientToken**
  Character. Unique, case-sensitive identifier that ensures the idempotency of the request. [optional]
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request...[optional]

InstanceCount Integer. The number of instances. Default: 1 [optional]

simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]

others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.

Value A list object or a character vector

LaunchSpecification

The launch specification. You must match the instance type, Availability Zone, network, and platform of the schedule that you purchased.

ScheduledInstanceId

The Scheduled Instance ID.

ClientToken

Unique, case-sensitive identifier that ensures the idempotency of the request. For more information, see Ensuring Idempotency.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

InstanceCount

The number of instances.

Default: 1
Description
Searches for routes in the specified local gateway route table.

Usage
ec2_search_local_gateway_routes(
  LocalGatewayRouteTableId,
  Filter,
  MaxResults = NULL,
  NextToken = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments
LocalGatewayRouteTableId
  Character. The ID of the local gateway route table.
Filter
  Named list where the name is the filter name and the value is the filter’s value.
  The value can be a vector or a list object (see below)[optional]
MaxResults
  Integer. The maximum number of results to return with a single call.[optional]
NextToken
  Characters. The token for the next page of results[optional]
DryRun
  Logical. Checks whether you have the required permissions for the action, without actually making the request....[optional]
simplify
  Logical. Whether to simplify the result and handle nextToken in the response[optional]
others
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error
  Logical. Whether to show an error message when a network error occurs.
retry_time
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region
  Character. The region of the AWS service.
Value

A list object or a character vector

LocalGatewayRouteTableId

The ID of the local gateway route table.

Filter

One or more filters.

MaxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

description

Search Transit Gateway Multicast Groups

Searches one or more transit gateway multicast groups and returns the group membership information.

Usage

ec2_search_transit_gateway_multicast_groups(
    TransitGatewayMulticastDomainId = NULL,
    Filter = NULL,
    MaxResults = NULL,
    NextToken = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
Arguments

**TransitGatewayMulticastDomainId**
Character. The ID of the transit gateway multicast domain.[optional]

**Filter**
Named list where the name is the filter name and the value is the filter's value. The value can be a vector or a list object (see below)[optional]

**MaxResults**
Integer. The maximum number of results to return with a single call.[optional]

**NextToken**
Characters. The token for the next page of results[optional]

**DryRun**
Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

**simplify**
Logical. Whether to simplify the result and handle nextToken in the response[optional]

**others**
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

**print_on_error**
Logical. Whether to show an error message when a network error occurs.

**retry_time**
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

**network_timeout**
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

**region**
Character. The region of the AWS service.

Value

A list object or a character vector

**TransitGatewayMulticastDomainId**
The ID of the transit gateway multicast domain.

**Filter**

One or more filters. The possible values are:

- **group-ip-address** - The IP address of the transit gateway multicast group.
- **is-group-member** - The resource is a group member. Valid values are true \| false.
- **is-group-source** - The resource is a group source. Valid values are true \| false.
- **member-type** - The member type. Valid values are igmp \| static.
- **resource-id** - The ID of the resource.
- **resource-type** - The type of resource. Valid values are vpc \| vpn \| direct-connect-gateway \| tgw-peering.
- **source-type** - The source type. Valid values are igmp \| static.
- **state** - The state of the subnet association. Valid values are associated \| associated \| disassociated \| disassociating.
- **subnet-id** - The ID of the subnet.
- **transit-gateway-attachment-id** - The id of the transit gateway attachment.
MaxResults

The maximum number of results to return with a single call. To retrieve the remaining results, make another call with the returned nextToken value.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Usage

```r
e2_search_transit_gateway_routes(
  TransitGatewayRouteTableId,
  Filter,
  MaxResults = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **TransitGatewayRouteTableId** Character. The ID of the transit gateway route table.
- **Filter** Named list where the name is the filter name and the value is the filter’s value. The value can be a vector or a list object (see below)[optional]
- **MaxResults** Integer. The maximum number of routes to return.[optional]
- **DryRun** Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
- **simplify** Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others** Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value
A list object or a character vector

TransitGatewayRouteTableId
The ID of the transit gateway route table.

Filter
One or more filters. The possible values are:
• attachment.transit-gateway-attachment-id - The id of the transit gateway attachment.
• attachment.resource-id - The resource id of the transit gateway attachment.
• attachment.resource-type - The attachment resource type. Valid values are vpc \ vpc \ direct-connect-gateway \ peering \ connect.
• prefix-list-id - The ID of the prefix list.
• route-search.exact-match - The exact match of the specified filter.
• route-search.longest-prefix-match - The longest prefix that matches the route.
• route-search.subnet-of-match - The routes with a subnet that match the specified CIDR filter.
• route-search.supernet-of-match - The routes with a CIDR that encompass the CIDR filter. For example, if you have 10.0.1.0/29 and 10.0.1.0/31 routes in your route table and you specify supernet-of-match as 10.0.1.0/30, then the result returns 10.0.1.0/29.
• state - The state of the route (active \ blackhole).
• type - The type of route (propagated \ static).

MaxResults
The maximum number of routes to return.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
ec2_send_diagnostic_interrupt

Send Diagnostic Interrupt

Description

Send Diagnostic Interrupt

Usage

ec2_send_diagnostic_interrupt(
    InstanceId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

InstanceId  Character. The ID of the instance.
DryRun      Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
simplify    Logical. Whether to simplify the result and handle nextToken in the response[optional]
others      Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region      Character. The region of the AWS service.

Value

A list object or a character vector

InstanceId

The ID of the instance.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response isDryRunOperation. Otherwise, it isUnauthorizedOperation.

ec2_start_instances  Start Instances

Description

Start Instances

Usage

ec2_start_instances(
  InstanceId,
  AdditionalInfo = NULL,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>InstanceId</td>
<td>List. The IDs of the instances.</td>
</tr>
<tr>
<td>AdditionalInfo</td>
<td>Character. Reserved.[optional]</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response[optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>
Value

A list object or a character vector

InstanceId

The IDs of the instances.

AdditionalInfo

Reserved.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

---

```r
ec2_start_network_insights_analysis

Start Network Insights Analysis

ec2_start_network_insights_analysis

    NetworkInsightsPathId,
    ClientToken,
    FilterInArn = NULL,
    DryRun = NULL,
    TagSpecification = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
```

Description

Starts analyzing the specified path. If the path is reachable, the operation returns the shortest feasible path.

Usage
Arguments

NetworkInsightsPathId
  Character. The ID of the path.

ClientToken
  Character. Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.

FilterInArn
  List. The Amazon Resource Names (ARN) of the resources that the path must traverse.[optional]

DryRun
  Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

TagSpecification
  List. The tags to apply.[optional]

simplify
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
  Logical. Whether to show an error message when a network error occurs.

retry_time
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
  Character. The region of the AWS service.

Value

A list object or a character vector

NetworkInsightsPathId

The ID of the path.

ClientToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. For more information, see How to Ensure Idempotency.

FilterInArn

The Amazon Resource Names (ARN) of the resources that the path must traverse.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
TagSpecification

The tags to apply.

ec2_start_vpc_endpoint_service_private_dns_verification

Start Vpc Endpoint Service Private Dns Verification

Description

Start Vpc Endpoint Service Private Dns Verification

Usage

```r
ec2_start_vpc_endpoint_service_private_dns_verification(
  ServiceId,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ServiceId</td>
<td>Character. The ID of the endpoint service.</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response[optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>

Value

A list object or a character vector
ServiceId

The ID of the endpoint service.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

description

Stop Instances

Usage

e2_stop_instances(
    InstanceId,
    Hibernate = NULL,
    DryRun = NULL,
    Force = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

InstanceId List. The IDs of the instances.
Hibernate Logical. Hibernates the instance if the instance was enabled for hibernation at launch.[optional]
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
Force Logical. Forces the instances to stop.[optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
Terminates active Client VPN endpoint connections. This action can be used to terminate a specific client connection, or up to five connections established by a specific user.
Usage

ec2_terminate_client_vpn_connections(
    ClientVpnEndpointId,
    ConnectionId = NULL,
    Username = NULL,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

ClientVpnEndpointId
    Character. The ID of the Client VPN endpoint to which the client is connected.
ConnectionId
    Character. The ID of the client connection to be terminated.[optional]
Username
    Character. The name of the user who initiated the connection.[optional]
DryRun
    Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
simplify
    Logical. Whether to simplify the result and handle nextToken in the response[optional]
others
    Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error
    Logical. Whether to show an error message when a network error occurs.
retry_time
    Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout
    Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region
    Character. The region of the AWS service.

Value

A list object or a character vector

ClientVpnEndpointId
    The ID of the Client VPN endpoint to which the client is connected.

ConnectionId
    The ID of the client connection to be terminated.
Username

The name of the user who initiated the connection. Use this option to terminate all active connections for the specified user. This option can only be used if the user has established up to five connections.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

description

Terminate Instances

Usage

`ec2_terminate_instances`

Arguments

```
InstanceId    List. The IDs of the instances.
DryRun        Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]
simplify      Logical. Whether to simplify the result and handle nextToken in the response[optional]
others        Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time    Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
```
Description
Unassigns one or more IPv6 addresses from a network interface.

Usage
ec2_unassign_ipv6_addresses(
    Ipv6Addresses,
    NetworkInterfaceId,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments
Ipv6Addresses List. The IPv6 addresses to unassign from the network interface.
NetworkInterfaceId Character. The ID of the network interface.
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others  

Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error  

Logical. Whether to show an error message when a network error occurs.

retry_time  

Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  

Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  

Character. The region of the AWS service.

Value  

A list object or a character vector

Ipv6Addresses  

The IPv6 addresses to unassign from the network interface.

NetworkInterfaceId  

The ID of the network interface.

decl

ec2_unassign_private_ip_addresses  

Unassign Private Ip Addresses

desc

Unassigns one or more secondary private IP addresses from a network interface.

usage

ec2_unassign_private_ip_addresses(  
  NetworkInterfaceId,  
  PrivateIpAddress,  
  simplify = TRUE,  
  others = list(),  
  print_on_error = aws_get_print_on_error(),  
  retry_time = aws_get_retry_time(),  
  network_timeout = aws_get_network_timeout(),  
  region = aws_get_region()  
)
ec2_unmonitor_instances

Arguments

NetworkInterfaceId Character. The ID of the network interface.

PrivateIpAddress List. The secondary private IP addresses to unassign from the network interface.

simplify Logical. Whether to simplify the result and handle nextToken in the response [optional]

others Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.

Value

A list object or a character vector

NetworkInterfaceId

The ID of the network interface.

PrivatelpIpAddress

The secondary private IP addresses to unassign from the network interface. You can specify this option multiple times to unassign more than one IP address.

Description

Disables detailed monitoring for a running instance. For more information, see Monitoring your instances and volumes in the Amazon EC2 User Guide.
Usage

ec2_unmonitor_instances(
    InstanceId,
    DryRun = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

InstanceId List. The IDs of the instances.
DryRun Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value

A list object or a character vector

InstanceId

The IDs of the instances.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.
ec2_update_security_group_rule_descriptions_egress

Update Security Group Rule Descriptions Egress

Description
Update Security Group Rule Descriptions Egress

Usage
ec2_update_security_group_rule_descriptions_egress(
  IpPermissions,
  DryRun = NULL,
  GroupId = NULL,
  GroupName = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments
IpPermissions  List. The IP permissions for the security group rule.
DryRun        Logical. Checks whether you have the required permissions for the action, without actually making the request.[optional]
GroupId        Character. The ID of the security group.[optional]
GroupName      Character. [Default VPC] The name of the security group.[optional]
simplify       Logical. Whether to simplify the result and handle nextToken in the response[optional]
others         Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time     Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region         Character. The region of the AWS service.

Value
A list object or a character vector
IpPermissions

The IP permissions for the security group rule.

DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

GroupId

The ID of the security group. You must specify either the security group ID or the security group name in the request. For security groups in a nondefault VPC, you must specify the security group ID.

GroupName

[Default VPC] The name of the security group. You must specify either the security group ID or the security group name in the request.

Usage

```
ec2_update_security_group_rule_descriptions_ingress(
    IpPermissions,
    DryRun = NULL,
    GroupId = NULL,
    GroupName = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```
Arguments

IpPermissions
List. The IP permissions for the security group rule.

DryRun
Logical. Checks whether you have the required permissions for the action, without actually making the request,...[optional]

GroupId
Character. The ID of the security group.[optional]

GroupName
Character. [EC2-Classic, default VPC] The name of the security group.[optional]

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value
A list object or a character vector

IpPermissions
The IP permissions for the security group rule.

DryRun
Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

GroupId
The ID of the security group. You must specify either the security group ID or the security group name in the request. For security groups in a nondefault VPC, you must specify the security group ID.

GroupName
[EC2-Classic, default VPC] The name of the security group. You must specify either the security group ID or the security group name in the request.
ec2_withdraw_byoip_cidr

Withdraw Byoip Cidr

Description

Withdraw Byoip Cidr

Usage

ec2_withdraw_byoip_cidr(
  Cidr,
  DryRun = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cidr</td>
<td>Character. The address range, in CIDR notation.</td>
</tr>
<tr>
<td>DryRun</td>
<td>Logical. Checks whether you have the required permissions for the action, without actually making the request. [optional]</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response. [optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request. [optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>

Value

A list object or a character vector

Cidr

The address range, in CIDR notation.
DryRun

Checks whether you have the required permissions for the action, without actually making the request, and provides an error response. If you have the required permissions, the error response is DryRunOperation. Otherwise, it is UnauthorizedOperation.

Description

Create a new capacity provider. Capacity providers are associated with an Amazon ECS cluster and are used in capacity provider strategies to facilitate cluster auto scaling.

Usage

```r
ecs_create_capacity_provider(
  name = NULL,
  autoScalingGroupProvider = NULL,
  tags = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **name** Character. The name of the capacity provider.
- **autoScalingGroupProvider** Object. The details of the Auto Scaling group for the capacity provider.
- **tags** List. The metadata that you apply to the capacity provider to help you categorize and organize them.
- **simplify** Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others** Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error** Logical. Whether to show an error message when a network error occurs.
- **retry_time** Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- **network_timeout** Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region** Character. The region of the AWS service.
Value

A list object or a character vector

name

The name of the capacity provider. Up to 255 characters are allowed, including letters (upper and lowercase), numbers, underscores, and hyphens. The name cannot be prefixed with 'aws', 'ecs', or 'fargate'.

autoScalingGroupProvider

The details of the Auto Scaling group for the capacity provider.

tags

The metadata that you apply to the capacity provider to help you categorize and organize them. Each tag consists of a key and an optional value, both of which you define.

The following basic restrictions apply to tags:

- Maximum number of tags per resource - 50
- For each resource, each tag key must be unique, and each tag key can have only one value.
- Maximum key length - 128 Unicode characters in UTF-8
- Maximum value length - 256 Unicode characters in UTF-8
- If your tagging schema is used across multiple services and resources, remember that other services may have restrictions on allowed characters. Generally allowed characters are: letters, numbers, and spaces representable in UTF-8, and the following characters: + - = . _ : / @.
- Tag keys and values are case-sensitive.
- Do not use aws:, AWS:, or any upper or lowercase combination of such as a prefix for either keys or values as it is reserved for AWS use. You cannot edit or delete tag keys or values with this prefix. Tags with this prefix do not count against your tags per resource limit.
Usage

ecs_create_cluster(
  clusterName = NULL,
  tags = NULL,
  settings = NULL,
  capacityProviders = NULL,
  defaultCapacityProviderStrategy = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

clusterName Character. The name of your cluster.
tags List. The metadata that you apply to the cluster to help you categorize and organize them.
settings List. The setting to use when creating a cluster.
capacityProviders List. The short name of one or more capacity providers to associate with the cluster.
defaultCapacityProviderStrategy List. The capacity provider strategy to use by default for the cluster.
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value

A list object or a character vector

clusterName

The name of your cluster. If you do not specify a name for your cluster, you create a cluster named default. Up to 255 letters (uppercase and lowercase), numbers, and hyphens are allowed.
tags

The metadata that you apply to the cluster to help you categorize and organize them. Each tag consists of a key and an optional value, both of which you define.

The following basic restrictions apply to tags:

- Maximum number of tags per resource - 50
- For each resource, each tag key must be unique, and each tag key can have only one value.
- Maximum key length - 128 Unicode characters in UTF-8
- Maximum value length - 256 Unicode characters in UTF-8
- If your tagging schema is used across multiple services and resources, remember that other services may have restrictions on allowed characters. Generally allowed characters are: letters, numbers, and spaces representable in UTF-8, and the following characters: + - = . _ : / @.
- Tag keys and values are case-sensitive.
- Do not use aws:,, AWS:, or any upper or lowercase combination of such as a prefix for either keys or values as it is reserved for AWS use. You cannot edit or delete tag keys or values with this prefix. Tags with this prefix do not count against your tags per resource limit.

settings

The setting to use when creating a cluster. This parameter is used to enable CloudWatch Container Insights for a cluster. If this value is specified, it will override the containerInsights value set with PutAccountSetting or PutAccountSettingDefault.

capacityProviders

The short name of one or more capacity providers to associate with the cluster.

If specifying a capacity provider that uses an Auto Scaling group, the capacity provider must already be created and not already associated with another cluster. New capacity providers can be created with the CreateCapacityProvider API operation.

To use a AWS Fargate capacity provider, specify either the FARGATE or FARGATE_SPOT capacity providers. The AWS Fargate capacity providers are available to all accounts and only need to be associated with a cluster to be used.

The PutClusterCapacityProviders API operation is used to update the list of available capacity providers for a cluster after the cluster is created.

defaultCapacityProviderStrategy

The capacity provider strategy to use by default for the cluster.

When creating a service or running a task on a cluster, if no capacity provider or launch type is specified then the default capacity provider strategy for the cluster is used.

A capacity provider strategy consists of one or more capacity providers along with the base and weight to assign to them. A capacity provider must be associated with the cluster to be used in a capacity provider strategy. The PutClusterCapacityProviders API is used to associate a capacity provider with a cluster. Only capacity providers with an ACTIVE or UPDATING status can be used.
If specifying a capacity provider that uses an Auto Scaling group, the capacity provider must already be created. New capacity providers can be created with the CreateCapacityProvider API operation.

To use an AWS Fargate capacity provider, specify either the FARGATE or FARGATE_SPOT capacity providers. The AWS Fargate capacity providers are available to all accounts and only need to be associated with a cluster to be used.

If a default capacity provider strategy is not defined for a cluster during creation, it can be defined later with the PutClusterCapacityProviders API operation.

---

**ecs_create_service**

**Create Service**

**Description**

Create Service

**Usage**

```r
ecs_create_service(
  cluster = NULL,
  serviceName = NULL,
  taskDefinition = NULL,
  loadBalancers = NULL,
  serviceRegistries = NULL,
  desiredCount = NULL,
  clientToken = NULL,
  launchType = NULL,
  capacityProviderStrategy = NULL,
  platformVersion = NULL,
  role = NULL,
  deploymentConfiguration = NULL,
  placementConstraints = NULL,
  placementStrategy = NULL,
  networkConfiguration = NULL,
  healthCheckGracePeriodSeconds = NULL,
  schedulingStrategy = NULL,
  deploymentController = NULL,
  tags = NULL,
  enableECSManagedTags = NULL,
  propagateTags = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
Arguments

cluster
Character. The short name or full Amazon Resource Name (ARN) of the cluster on which to run your service.

serviceName
Character. The name of your service.

taskDefinition
Character. The family and revision (family:revision) or full ARN of the task definition to run in your...

loadBalancers
List. A load balancer object representing the load balancers to use with your service.

serviceRegistries
List. The details of the service discovery registries to assign to this service.

desiredCount
Integer. The number of instantiations of the specified task definition to place and keep running on your...

clientToken
Character. Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.

launchType
Character. The launch type on which to run your service.

capacityProviderStrategy
List. The capacity provider strategy to use for the service.

platformVersion
Character. The platform version that your tasks in the service are running on.

role
Character. The name or full Amazon Resource Name (ARN) of the IAM role that allows Amazon ECS to make calls...

deploymentConfiguration
Object. Optional deployment parameters that control how many tasks run during the deployment and the ordering...

placementConstraints
List. An array of placement constraint objects to use for tasks in your service.

placementStrategy
List. The placement strategy objects to use for tasks in your service.

networkConfiguration
Object. The network configuration for the service.

healthCheckGracePeriodSeconds
Integer. The period of time, in seconds, that the Amazon ECS service scheduler should ignore unhealthy Elastic...

schedulingStrategy
Character. The scheduling strategy to use for the service.

deploymentController
Object. The deployment controller to use for the service.

tags
List. The metadata that you apply to the service to help you categorize and organize them.

enableECSManagedTags
Logical. Specifies whether to enable Amazon ECS managed tags for the tasks within the service.

propagateTags
Character. Specifies whether to propagate the tags from the task definition or the service to the tasks in the...
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]

others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.

Value
A list object or a character vector

cluster
The short name or full Amazon Resource Name (ARN) of the cluster on which to run your service. If you do not specify a cluster, the default cluster is assumed.

serviceName
The name of your service. Up to 255 letters (uppercase and lowercase), numbers, and hyphens are allowed. Service names must be unique within a cluster, but you can have similarly named services in multiple clusters within a Region or across multiple Regions.

taskDefinition
The family and revision (family:revision) or full ARN of the task definition to run in your service. If a revision is not specified, the latest ACTIVE revision is used.

A task definition must be specified if the service is using either the ECS or CODE_DEPLOY deployment controllers.

loadBalancers
A load balancer object representing the load balancers to use with your service. For more information, see Service Load Balancing in the Amazon Elastic Container Service Developer Guide.

If the service is using the rolling update (ECS) deployment controller and using either an Application Load Balancer or Network Load Balancer, you must specify one or more target group ARNs to attach to the service. The service-linked role is required for services that make use of multiple target groups. For more information, see Using Service-Linked Roles for Amazon ECS in the Amazon Elastic Container Service Developer Guide.

If the service is using the CODE_DEPLOY deployment controller, the service is required to use either an Application Load Balancer or Network Load Balancer. When creating an AWS CodeDeploy deployment group, you specify two target groups (referred to as a targetGroupPair). During a deployment, AWS CodeDeploy determines which task set in your service has the status PRIMARY and
associates one target group with it, and then associates the other target group with the replacement task set. The load balancer can also have up to two listeners: a required listener for production traffic and an optional listener that allows you perform validation tests with Lambda functions before routing production traffic to it.

After you create a service using the ECS deployment controller, the load balancer name or target group ARN, container name, and container port specified in the service definition are immutable. If you are using the CODE_DEPLOY deployment controller, these values can be changed when updating the service.

For Application Load Balancers and Network Load Balancers, this object must contain the load balancer target group ARN, the container name (as it appears in a container definition), and the container port to access from the load balancer. The load balancer name parameter must be omitted. When a task from this service is placed on a container instance, the container instance and port combination is registered as a target in the target group specified here.

For Classic Load Balancers, this object must contain the load balancer name, the container name (as it appears in a container definition), and the container port to access from the load balancer. The target group ARN parameter must be omitted. When a task from this service is placed on a container instance, the container instance is registered with the load balancer specified here.

Services with tasks that use the awsvpc network mode (for example, those with the Fargate launch type) only support Application Load Balancers and Network Load Balancers. Classic Load Balancers are not supported. Also, when you create any target groups for these services, you must choose ip as the target type, not instance, because tasks that use the awsvpc network mode are associated with an elastic network interface, not an Amazon EC2 instance.

**serviceRegistries**

The details of the service discovery registries to assign to this service. For more information, see Service Discovery.

Service discovery is supported for Fargate tasks if you are using platform version v1.1.0 or later. For more information, see AWS Fargate Platform Versions.

**desiredCount**

The number of instantiations of the specified task definition to place and keep running on your cluster.

This is required if schedulingStrategy is REPLICA or is not specified. If schedulingStrategy is DAEMON then this is not required.

**clientToken**

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. Up to 32 ASCII characters are allowed.

**launchType**

The launch type on which to run your service. For more information, see Amazon ECS Launch Types in the Amazon Elastic Container Service Developer Guide.

If a launchType is specified, the capacityProviderStrategy parameter must be omitted.
**capacityProviderStrategy**

The capacity provider strategy to use for the service.

A capacity provider strategy consists of one or more capacity providers along with the base and weight to assign to them. A capacity provider must be associated with the cluster to be used in a capacity provider strategy. The PutClusterCapacityProviders API is used to associate a capacity provider with a cluster. Only capacity providers with an ACTIVE or UPDATING status can be used.

If a capacityProviderStrategy is specified, the launchType parameter must be omitted. If no capacityProviderStrategy or launchType is specified, the defaultCapacityProviderStrategy for the cluster is used.

If specifying a capacity provider that uses an Auto Scaling group, the capacity provider must already be created. New capacity providers can be created with the CreateCapacityProvider API operation.

To use a AWS Fargate capacity provider, specify either the FARGATE or FARGATE_SPOT capacity providers. The AWS Fargate capacity providers are available to all accounts and only need to be associated with a cluster to be used.

The PutClusterCapacityProviders API operation is used to update the list of available capacity providers for a cluster after the cluster is created.

**platformVersion**

The platform version that your tasks in the service are running on. A platform version is specified only for tasks using the Fargate launch type. If one isn't specified, the LATEST platform version is used by default. For more information, see AWS Fargate Platform Versions in the Amazon Elastic Container Service Developer Guide.

**role**

The name or full Amazon Resource Name (ARN) of the IAM role that allows Amazon ECS to make calls to your load balancer on your behalf. This parameter is only permitted if you are using a load balancer with your service and your task definition does not use the awsvpc network mode. If you specify the role parameter, you must also specify a load balancer object with the loadBalancers parameter.

If your account has already created the Amazon ECS service-linked role, that role is used by default for your service unless you specify a role here. The service-linked role is required if your task definition uses the awsvpc network mode or if the service is configured to use service discovery, an external deployment controller, multiple target groups, or Elastic Inference accelerators in which case you should not specify a role here. For more information, see Using Service-Linked Roles for Amazon ECS in the Amazon Elastic Container Service Developer Guide.

If your specified role has a path other than /, then you must either specify the full role ARN (this is recommended) or prefix the role name with the path. For example, if a role with the name bar has a path of /foo/ then you would specify /foo/bar as the role name. For more information, see Friendly Names and Paths in the IAM User Guide.

**deploymentConfiguration**

Optional deployment parameters that control how many tasks run during the deployment and the ordering of stopping and starting tasks.
placementConstraints

An array of placement constraint objects to use for tasks in your service. You can specify a maximum of 10 constraints per task (this limit includes constraints in the task definition and those specified at runtime).

placementStrategy

The placement strategy objects to use for tasks in your service. You can specify a maximum of five strategy rules per service.

taskGroup

An array of task group objects to use for tasks in your service. You can specify a maximum of ten constraints per task (this limit includes constraints in the task definition and those specified at runtime).

networkConfiguration

The network configuration for the service. This parameter is required for task definitions that use the awsvpc network mode to receive their own elastic network interface, and it is not supported for other network modes. For more information, see Task Networking in the Amazon Elastic Container Service Developer Guide.

healthCheckGracePeriodSeconds

The period of time, in seconds, that the Amazon ECS service scheduler should ignore unhealthy Elastic Load Balancing target health checks after a task has first started. This is only used when your service is configured to use a load balancer. If your service has a load balancer defined and you don’t specify a health check grace period value, the default value of 0 is used.

If your service’s tasks take a while to start and respond to Elastic Load Balancing health checks, you can specify a health check grace period of up to 2,147,483,647 seconds. During that time, the Amazon ECS service scheduler ignores health check status. This grace period can prevent the service scheduler from marking tasks as unhealthy and stopping them before they have time to come up.

schedulingStrategy

The scheduling strategy to use for the service. For more information, see Services.

There are two service scheduler strategies available:

- REPLICA-The replica scheduling strategy places and maintains the desired number of tasks across your cluster. By default, the service scheduler spreads tasks across Availability Zones. You can use task placement strategies and constraints to customize task placement decisions. This scheduler strategy is required if the service is using the CODE_DEPLOY or EXTERNAL deployment controller types.

- DAEMON-The daemon scheduling strategy deploys exactly one task on each active container instance that meets all of the task placement constraints that you specify in your cluster. The service scheduler also evaluates the task placement constraints for running tasks and will stop tasks that do not meet the placement constraints. When you’re using this strategy, you don’t need to specify a desired number of tasks, a task placement strategy, or use Service Auto Scaling policies.

Tasks using the Fargate launch type or the CODE_DEPLOY or EXTERNAL deployment controller types don’t support the DAEMON scheduling strategy.
deploymentController

The deployment controller to use for the service.

tags

The metadata that you apply to the service to help you categorize and organize them. Each tag consists of a key and an optional value, both of which you define. When a service is deleted, the tags are deleted as well.

The following basic restrictions apply to tags:

- Maximum number of tags per resource - 50
- For each resource, each tag key must be unique, and each tag key can have only one value.
- Maximum key length - 128 Unicode characters in UTF-8
- Maximum value length - 256 Unicode characters in UTF-8
- If your tagging schema is used across multiple services and resources, remember that other services may have restrictions on allowed characters. Generally allowed characters are: letters, numbers, and spaces representable in UTF-8, and the following characters: + - . _ : / @.
- Tag keys and values are case-sensitive.
- Do not use aws:, AWS:, or any upper or lowercase combination of such as a prefix for either keys or values as it is reserved for AWS use. You cannot edit or delete tag keys or values with this prefix. Tags with this prefix do not count against your tags per resource limit.

enableECSManagedTags

Specifies whether to enable Amazon ECS managed tags for the tasks within the service. For more information, see Tagging Your Amazon ECS Resources in the Amazon Elastic Container Service Developer Guide.

propagateTags

Specifies whether to propagate the tags from the task definition or the service to the tasks in the service. If no value is specified, the tags are not propagated. Tags can only be propagated to the tasks within the service during service creation. To add tags to a task after service creation, use the TagResource API action.

---

c_create_task_set

Create Task Set

Description

Create a task set in the specified cluster and service. This is used when a service uses the EXTERNAL deployment controller type. For more information, see Amazon ECS Deployment Types in the Amazon Elastic Container Service Developer Guide.
Usage

ecs_create_task_set(
    service = NULL,
    cluster = NULL,
    externalId = NULL,
    taskDefinition = NULL,
    networkConfiguration = NULL,
    loadBalancers = NULL,
    serviceRegistries = NULL,
    launchType = NULL,
    capacityProviderStrategy = NULL,
    platformVersion = NULL,
    scale = NULL,
    clientToken = NULL,
    tags = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

service    Character. The short name or full Amazon Resource Name (ARN) of the service to create the task set in.
cluster    Character. The short name or full Amazon Resource Name (ARN) of the cluster that hosts the service to create...
externalId Character. An optional non-unique tag that identifies this task set in external systems.
taskDefinition Character. The task definition for the tasks in the task set to use.
networkConfiguration No description can be found.
loadBalancers List. A load balancer object representing the load balancer to use with the task set.
serviceRegistries List. The details of the service discovery registries to assign to this task set.
launchType   Character. The launch type that new tasks in the task set will use.
capacityProviderStrategy List. The capacity provider strategy to use for the task set.
platformVersion Character. The platform version that the tasks in the task set should use.
scale        No description can be found.
clientToken  Character. Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.
tags
List. The metadata that you apply to the task set to help you categorize and organize them.
simplify
Logical. Whether to simplify the result and handle nextToken in the response (optional)
others
Named list. The parameters that are not included in the function parameters and need to be added into the request (optional)
print_on_error
Logical. Whether to show an error message when a network error occurs.
retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region
Character. The region of the AWS service.

Value
A list object or a character vector

service
The short name or full Amazon Resource Name (ARN) of the service to create the task set in.

cluster
The short name or full Amazon Resource Name (ARN) of the cluster that hosts the service to create the task set in.

externalId
An optional non-unique tag that identifies this task set in external systems. If the task set is associated with a service discovery registry, the tasks in this task set will have the ECS_TASK_SET_EXTERNAL_ID AWS Cloud Map attribute set to the provided value.

taskDefinition
The task definition for the tasks in the task set to use.

networkConfiguration
No description can be found.

loadBalancers
A load balancer object representing the load balancer to use with the task set. The supported load balancer types are either an Application Load Balancer or a Network Load Balancer.
serviceRegistries

The details of the service discovery registries to assign to this task set. For more information, see Service Discovery.

launchType

The launch type that new tasks in the task set will use. For more information, see Amazon ECS Launch Types in the Amazon Elastic Container Service Developer Guide.

If a launchType is specified, the capacityProviderStrategy parameter must be omitted.

capacityProviderStrategy

The capacity provider strategy to use for the task set.

A capacity provider strategy consists of one or more capacity providers along with the base and weight to assign to them. A capacity provider must be associated with the cluster to be used in a capacity provider strategy. The PutClusterCapacityProviders API is used to associate a capacity provider with a cluster. Only capacity providers with an ACTIVE or UPDATING status can be used.

If a capacityProviderStrategy is specified, the launchType parameter must be omitted. If no capacityProviderStrategy or launchType is specified, the defaultCapacityProviderStrategy for the cluster is used.

If specifying a capacity provider that uses an Auto Scaling group, the capacity provider must already be created. New capacity providers can be created with the CreateCapacityProvider API operation.

To use a AWS Fargate capacity provider, specify either the FARGATE or FARGATE_SPOT capacity providers. The AWS Fargate capacity providers are available to all accounts and only need to be associated with a cluster to be used.

The PutClusterCapacityProviders API operation is used to update the list of available capacity providers for a cluster after the cluster is created.

platformVersion

The platform version that the tasks in the task set should use. A platform version is specified only for tasks using the Fargate launch type. If one isn’t specified, the LATEST platform version is used by default.

scale

No description can be found.

clientToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request. Up to 32 ASCII characters are allowed.
tags

The metadata that you apply to the task set to help you categorize and organize them. Each tag consists of a key and an optional value, both of which you define. When a service is deleted, the tags are deleted as well.

The following basic restrictions apply to tags:

- Maximum number of tags per resource - 50
- For each resource, each tag key must be unique, and each tag key can have only one value.
- Maximum key length - 128 Unicode characters in UTF-8
- Maximum value length - 256 Unicode characters in UTF-8
- If your tagging schema is used across multiple services and resources, remember that other services may have restrictions on allowed characters. Generally allowed characters are: letters, numbers, and spaces representable in UTF-8, and the following characters: + - = . _ : / @.
- Tag keys and values are case-sensitive.
- Do not use `aws:` or `AWS:` or any uppercase or lowercase combination of such as a prefix for either keys or values as it is reserved for AWS use. You cannot edit or delete tag keys or values with this prefix. Tags with this prefix do not count against your tags per resource limit.

---

**ecs_delete_account_setting**

*Delete Account Setting*

---

**Description**

Disables an account setting for a specified IAM user, IAM role, or the root user for an account.

**Usage**

```plaintext
code
ecs_delete_account_setting(
  name = NULL,
  principalArn = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- `name` Character. The resource name for which to disable the account setting.
- `principalArn` Character. The ARN of the principal, which can be an IAM user, IAM role, or the root user.
Description

Deletes one or more custom attributes from an Amazon ECS resource.

Usage

```r
ecs_delete_attributes(
    cluster = NULL,
    attributes = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
```
ecs_delete_capacity_provider

retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()

Arguments

cluster  Character. The short name or full Amazon Resource Name (ARN) of the cluster that contains the resource to delete...

attributes  List. The attributes to delete from your resource.

simplify  Logical. Whether to simplify the result and handle nextToken in the response[optional]

others  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error  Logical. Whether to show an error message when a network error occurs.

retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  Character. The region of the AWS service.

Value

A list object or a character vector

cluster

The short name or full Amazon Resource Name (ARN) of the cluster that contains the resource to delete attributes. If you do not specify a cluster, the default cluster is assumed.

attributes

The attributes to delete from your resource. You can specify up to 10 attributes per request. For custom attributes, specify the attribute name and target ID, but do not specify the value. If you specify the target ID using the short form, you must also specify the target type.

Description

Delete Capacity Provider
Usage

ecs_delete_capacity_provider(
    capacityProvider = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

capacityProvider
    Character. The short name or full Amazon Resource Name (ARN) of the capacity provider to delete.
simplify
    Logical. Whether to simplify the result and handle nextToken in the response[optional]
others
    Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error
    Logical. Whether to show an error message when a network error occurs.
retry_time
    Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout
    Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region
    Character. The region of the AWS service.

Value

A list object or a character vector

capacityProvider
    The short name or full Amazon Resource Name (ARN) of the capacity provider to delete.

Description

Delete Cluster
Usage

ecs_delete_cluster(
    cluster = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

cluster  Character. The short name or full Amazon Resource Name (ARN) of the cluster
to delete.

simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and
need to be added into the request[optional]

print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network
issue. If the request has been sent retry_time times but still not be able to get
the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can
not be less than 1 ms.
region Character. The region of the AWS service.

Value

A list object or a character vector

c Cluster

The short name or full Amazon Resource Name (ARN) of the cluster to delete.

Description

Delete Service
Usage

ecs_delete_service(
  cluster = NULL,
  service = NULL,
  force = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

cluster Character. The short name or full Amazon Resource Name (ARN) of the cluster that hosts the service to delete.

service Character. The name of the service to delete.

force Logical. If true, allows you to delete a service even if it has not been scaled down to zero tasks.

simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]

others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.

Value

A list object or a character vector

cluster

The short name or full Amazon Resource Name (ARN) of the cluster that hosts the service to delete. If you do not specify a cluster, the default cluster is assumed.

service

The name of the service to delete.
force

If true, allows you to delete a service even if it has not been scaled down to zero tasks. It is only necessary to use this if the service is using the REPLICA scheduling strategy.

Description

Deletes a specified task set within a service. This is used when a service uses the EXTERNAL deployment controller type. For more information, see Amazon ECS Deployment Types in the Amazon Elastic Container Service Developer Guide.

Usage

```r
ecs_delete_task_set(
  cluster = NULL,
  service = NULL,
  taskSet = NULL,
  force = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **cluster**: Character. The short name or full Amazon Resource Name (ARN) of the cluster that hosts the service that the...
- **service**: Character. The short name or full Amazon Resource Name (ARN) of the service that hosts the task set to delete.
- **taskSet**: Character. The task set ID or full Amazon Resource Name (ARN) of the task set to delete.
- **force**: Logical. If true, this allows you to delete a task set even if it hasn’t been scaled down to zero.
- **simplify**: Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout
   Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
   Character. The region of the AWS service.

Value
   A list object or a character vector

cluster
   The short name or full Amazon Resource Name (ARN) of the cluster that hosts the service that the task set exists in to delete.

service
   The short name or full Amazon Resource Name (ARN) of the service that hosts the task set to delete.

taskSet
   The task set ID or full Amazon Resource Name (ARN) of the task set to delete.

force
   If true, this allows you to delete a task set even if it hasn’t been scaled down to zero.


desc

Description
   Deregister Container Instance

Usage

ecs_deregister_container_instance(
   cluster = NULL,
   containerInstance = NULL,
   force = NULL,
   simplify = TRUE,
   others = list(),
   print_on_error = aws_get_print_on_error(),
   retry_time = aws_get_retry_time(),
   network_timeout = aws_get_network_timeout(),
   region = aws_get_region()
)
**Arguments**

- **cluster**: Character. The short name or full Amazon Resource Name (ARN) of the cluster that hosts the container instance.
- **containerInstance**: Character. The container instance ID or full ARN of the container instance to deregister.
- **force**: Logical. Forces the deregistration of the container instance.
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response [optional].
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request [optional].
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

**Value**

A list object or a character vector

- **cluster**: The short name or full Amazon Resource Name (ARN) of the cluster that hosts the container instance to deregister. If you do not specify a cluster, the default cluster is assumed.

- **containerInstance**: The container instance ID or full ARN of the container instance to deregister. The ARN contains the `arn:aws:ecs` namespace, followed by the Region of the container instance, the AWS account ID of the container instance owner, the container-instance namespace, and then the container instance ID. For example, `arn:aws:ecs:region:aws_account_id:container-instance/container_instance_ID`.

- **force**: Forces the deregistration of the container instance. If you have tasks running on the container instance when you deregister it with the `force` option, these tasks remain running until you terminate the instance or the tasks stop through some other means, but they are orphaned (no longer monitored or accounted for by Amazon ECS). If an orphaned task on your container instance is part of an Amazon ECS service, then the service scheduler starts another copy of that task, on a different container instance if possible.

Any containers in orphaned service tasks that are registered with a Classic Load Balancer or an Application Load Balancer target group are deregistered. They begin connection draining according to the settings on the load balancer or target group.
**ecs_deregister_task_definition**

*Deregister Task Definition*

**Description**

Deregister Task Definition

**Usage**

```r
ecs_deregister_task_definition(
  taskDefinition = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- `taskDefinition`  Character. The family and revision (family:revision) or full Amazon Resource Name (ARN) of the task definition.
- `simplify`        Logical. Whether to simplify the result and handle nextToken in the response.[optional]
- `others`          Named list. The parameters that are not included in the function parameters and need to be added into the request.[optional]
- `print_on_error`  Logical. Whether to show an error message when a network error occurs.
- `retry_time`      Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- `network_timeout` Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- `region`          Character. The region of the AWS service.

**Value**

A list object or a character vector

**taskDefinition**

The family and revision (family:revision) or full Amazon Resource Name (ARN) of the task definition to deregister. You must specify a revision.
**Description**

Describes one or more of your capacity providers.

**Usage**

```r
ecs_describe_capacity_providers(
  capacityProviders = NULL,
  include = NULL,
  maxResults = NULL,
  nextToken = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- `capacityProviders`  
  List. The short name or full Amazon Resource Name (ARN) of one or more capacity providers.

- `include`  
  List. Specifies whether or not you want to see the resource tags for the capacity provider.

- `maxResults`  
  Integer. The maximum number of account setting results returned by `DescribeCapacityProviders` in paginated...

- `nextToken`  
  Characters. The token for the next page of results[optional]

- `simplify`  
  Logical. Whether to simplify the result and handle `nextToken` in the response[optional]

- `others`  
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

- `print_on_error`  
  Logical. Whether to show an error message when a network error occurs.

- `retry_time`  
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

- `network_timeout`  
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- `region`  
  Character. The region of the AWS service.
Value

A list object or a character vector

capacityProviders

The short name or full Amazon Resource Name (ARN) of one or more capacity providers. Up to 100 capacity providers can be described in an action.

include

Specifies whether or not you want to see the resource tags for the capacity provider. If TAGS is specified, the tags are included in the response. If this field is omitted, tags are not included in the response.

maxResults

The maximum number of account setting results returned by DescribeCapacityProviders in paginated output. When this parameter is used, DescribeCapacityProviders only returns maxResults results in a single page along with a nextToken response element. The remaining results of the initial request can be seen by sending another DescribeCapacityProviders request with the returned nextToken value. This value can be between 1 and 10. If this parameter is not used, then DescribeCapacityProviders returns up to 10 results and a nextToken value if applicable.

---

ecs_describe_clusters  Describe Clusters

Description

Describes one or more of your clusters.

Usage

```r
cs_describe_clusters(
  clusters = NULL,
  include = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
**Arguments**

- **clusters**: List. A list of up to 100 cluster names or full cluster Amazon Resource Name (ARN) entries.
- **include**: List. Whether to include additional information about your clusters in the response.
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response.[optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request.[optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

**Value**

A list object or a character vector

**clusters**

A list of up to 100 cluster names or full cluster Amazon Resource Name (ARN) entries. If you do not specify a cluster, the default cluster is assumed.

**include**

Whether to include additional information about your clusters in the response. If this field is omitted, the attachments, statistics, and tags are not included. If ATTACHMENTS is specified, the attachments for the container instances or tasks within the cluster are included. If SETTINGS is specified, the settings for the cluster are included. If STATISTICS is specified, the following additional information, separated by launch type, is included:

- `runningEC2TasksCount`
- `runningFargateTasksCount`
- `pendingEC2TasksCount`
- `pendingFargateTasksCount`
- `activeEC2ServiceCount`
- `activeFargateServiceCount`
- `drainingEC2ServiceCount`
- `drainingFargateServiceCount`

If TAGS is specified, the metadata tags associated with the cluster are included.
ecs_describe_container_instances

Describe Container Instances

Description

Describes Amazon Elastic Container Service container instances. Returns metadata about registered and remaining resources on each container instance requested.

Usage

ecs_describe_container_instances(
    cluster = NULL,
    containerInstances = NULL,
    include = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

cluster Character. The short name or full Amazon Resource Name (ARN) of the cluster that hosts the container instances...
containerInstances List. A list of up to 100 container instance IDs or full Amazon Resource Name (ARN) entries.
include List. Specifies whether you want to see the resource tags for the container instance.
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.
Value

A list object or a character vector

cluster

The short name or full Amazon Resource Name (ARN) of the cluster that hosts the container instances to describe. If you do not specify a cluster, the default cluster is assumed. This parameter is required if the container instance or container instances you are describing were launched in any cluster other than the default cluster.

containerInstances

A list of up to 100 container instance IDs or full Amazon Resource Name (ARN) entries.

include

Specifies whether you want to see the resource tags for the container instance. If TAGS is specified, the tags are included in the response. If this field is omitted, tags are not included in the response.

ecs_describe_services

Describe Services

Description

Describes the specified services running in your cluster.

Usage

```r
ecs_describe_services(
  cluster = NULL,
  services = NULL,
  include = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cluster</td>
<td>Character. The short name or full Amazon Resource Name (ARN) of the cluster that hosts the service to describe.</td>
</tr>
<tr>
<td>services</td>
<td>List. A list of services to describe.</td>
</tr>
<tr>
<td>include</td>
<td>List. Specifies whether you want to see the resource tags for the service.</td>
</tr>
</tbody>
</table>
**Description**

Describes a specified task or tasks.

**Usage**

```r
ces_describe_tasks(
    cluster = NULL,
    tasks = NULL,
    include = NULL,
    simplify = TRUE,
    others = list(),
)```
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

cluster Character. The short name or full Amazon Resource Name (ARN) of the cluster that hosts the task or tasks to...
tasks List. A list of up to 100 task IDs or full ARN entries.
include List. Specifies whether you want to see the resource tags for the task.
simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]
others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value

A list object or a character vector

cluster

The short name or full Amazon Resource Name (ARN) of the cluster that hosts the task or tasks to describe. If you do not specify a cluster, the default cluster is assumed. This parameter is required if the task or tasks you are describing were launched in any cluster other than the default cluster.

tasks

A list of up to 100 task IDs or full ARN entries.

include

Specifies whether you want to see the resource tags for the task. If TAGS is specified, the tags are included in the response. If this field is omitted, tags are not included in the response.
Description

Describe Task Definition

Usage

```r
escs_describe_task_definition(
  taskDefinition = NULL,
  include = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **taskDefinition**: Character. The family for the latest ACTIVE revision, family and revision (family:revision) for a specific revision in the family, or full Amazon Resource Name (ARN) of the task definition to describe.
- **include**: List. Specifies whether to see the resource tags for the task definition.
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response (optional).
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request (optional).
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

Value

A list object or a character vector

**taskDefinition**

The family for the latest ACTIVE revision, family and revision (family:revision) for a specific revision in the family, or full Amazon Resource Name (ARN) of the task definition to describe.
include

Specifies whether to see the resource tags for the task definition. If TAGS is specified, the tags are included in the response. If this field is omitted, tags are not included in the response.

Description

Describes the task sets in the specified cluster and service. This is used when a service uses the EXTERNAL deployment controller type. For more information, see Amazon ECS Deployment Types in the Amazon Elastic Container Service Developer Guide.

Usage

```r
ecs_describe_task_sets(
  cluster = NULL,
  service = NULL,
  taskSets = NULL,
  include = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **cluster** Character. The short name or full Amazon Resource Name (ARN) of the cluster that hosts the service that the...
- **service** Character. The short name or full Amazon Resource Name (ARN) of the service that the task sets exist in.
- **taskSets** List. The ID or full Amazon Resource Name (ARN) of task sets to describe.
- **include** List. Specifies whether to see the resource tags for the task set.
- **simplify** Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others** Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error** Logical. Whether to show an error message when a network error occurs.
- **retry_time** Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
**network_timeout**

Numeric. Number of seconds to wait for a REST response until giving up. Cannot be less than 1 ms.

**region**

Character. The region of the AWS service.

**Value**

A list object or a character vector

**cluster**

The short name or full Amazon Resource Name (ARN) of the cluster that hosts the service that the task sets exist in.

**service**

The short name or full Amazon Resource Name (ARN) of the service that the task sets exist in.

**taskSets**

The ID or full Amazon Resource Name (ARN) of task sets to describe.

**include**

Specifies whether to see the resource tags for the task set. If TAGS is specified, the tags are included in the response. If this field is omitted, tags are not included in the response.

---

**Description**

Discover Poll Endpoint

**Usage**

```r
ecs_discover_poll_endpoint(
  containerInstance = NULL,
  cluster = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
Arguments

containerInstance
Character. The container instance ID or full ARN of the container instance.

cluster
Character. The short name or full Amazon Resource Name (ARN) of the cluster to which the container instance...

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value

A list object or a character vector

containerInstance

The container instance ID or full ARN of the container instance. The ARN contains the arn:aws:ecs namespace, followed by the Region of the container instance, the AWS account ID of the container instance owner, the container-instance namespace, and then the container instance ID. For example, arn:aws:ecs:region:aws_account_id:container-instance/container_instance_ID.

cluster

The short name or full Amazon Resource Name (ARN) of the cluster to which the container instance belongs.

Description

Lists the account settings for a specified principal.
Usage

escs_list_account_settings(
    maxResults = NULL,
    nextToken = NULL,
    name = NULL,
    value = NULL,
    principalArn = NULL,
    effectiveSettings = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

maxResults  Character. Pagination limit[optional]
nextToken   Characters. The token for the next page of results[optional]
name        Character. The name of the account setting you want to list the settings for.
value       Character. The value of the account settings with which to filter results.
principalArn Character. The ARN of the principal, which can be an IAM user, IAM role, or the root user.
effectiveSettings Logical. Specifies whether to return the effective settings.
simplify    Logical. Whether to simplify the result and handle nextToken in the response[optional]
others      Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region      Character. The region of the AWS service.

Value

A list object or a character vector

maxResults

Pagination limit
name

The name of the account setting you want to list the settings for.

value

The value of the account settings with which to filter results. You must also specify an account setting name to use this parameter.

principalArn

The ARN of the principal, which can be an IAM user, IAM role, or the root user. If this field is omitted, the account settings are listed only for the authenticated user.

effectiveSettings

Specifies whether to return the effective settings. If true, the account settings for the root user or the default setting for the principalArn are returned. If false, the account settings for the principalArn are returned if they are set. Otherwise, no account settings are returned.

---

**Description**

Lists the attributes for Amazon ECS resources within a specified target type and cluster. When you specify a target type and cluster, ListAttributes returns a list of attribute objects, one for each attribute on each resource. You can filter the list of results to a single attribute name to only return results that have that name. You can also filter the results by attribute name and value, for example, to see which container instances in a cluster are running a Linux AMI (ecs.os-type=linux).

**Usage**

```bash
ecs_list_attributes(
    maxResults = NULL,
    nextToken = NULL,
    cluster = NULL,
    targetType = NULL,
    attributeName = NULL,
    attributeValue = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```
Arguments
maxResults  Character. Pagination limit[optional]
nextToken   Characters. The token for the next page of results[optional]
cluster     Character. The short name or full Amazon Resource Name (ARN) of the cluster to list attributes.
targetType  Character. The type of the target with which to list attributes.
attributeName  Character. The name of the attribute with which to filter the results.
attributeValue Character. The value of the attribute with which to filter results.
simplify    Logical. Whether to simplify the result and handle nextToken in the response[optional]
others      Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region      Character. The region of the AWS service.

Value
A list object or a character vector

maxResults
Pagination limit

cluster
The short name or full Amazon Resource Name (ARN) of the cluster to list attributes. If you do not specify a cluster, the default cluster is assumed.

targetType
The type of the target with which to list attributes.

attributeName
The name of the attribute with which to filter the results.

attributeValue
The value of the attribute with which to filter results. You must also specify an attribute name to use this parameter.
Description

Returns a list of existing clusters.

Usage

```
ecs_list_clusters(
    maxResults = NULL,
    nextToken = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

- `maxResults`: Character. Pagination limit[optional]
- `nextToken`: Characters. The token for the next page of results[optional]
- `simplify`: Logical. Whether to simplify the result and handle `nextToken` in the response[optional]
- `others`: Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- `print_on_error`: Logical. Whether to show an error message when a network error occurs.
- `retry_time`: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- `network_timeout`: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- `region`: Character. The region of the AWS service.

Value

A list object or a character vector

**maxResults**

Pagination limit
ecs_list_container_instances

*List Container Instances*

**Description**

Returns a list of container instances in a specified cluster. You can filter the results of a `ListContainerInstances` operation with cluster query language statements inside the `filter` parameter. For more information, see Cluster Query Language in the Amazon Elastic Container Service Developer Guide.

**Usage**

```r
ecs_list_container_instances(
  maxResults = NULL,
  nextToken = NULL,
  cluster = NULL,
  filter = NULL,
  status = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- `maxResults` Character. Pagination limit[optional]
- `nextToken` Characters. The token for the next page of results[optional]
- `cluster` Character. The short name or full Amazon Resource Name (ARN) of the cluster that hosts the container instances...
- `filter` Character. You can filter the results of a `ListContainerInstances` operation with cluster query language statements....
- `status` Character. Filters the container instances by status.
- `simplify` Logical. Whether to simplify the result and handle `nextToken` in the response[optional]
- `others` Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- `print_on_error` Logical. Whether to show an error message when a network error occurs.
- `retry_time` Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- `network_timeout` Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- `region` Character. The region of the AWS service.
Value
A list object or a character vector

maxResults
Pagination limit

cluster
The short name or full Amazon Resource Name (ARN) of the cluster that hosts the container instances to list. If you do not specify a cluster, the default cluster is assumed.

filter
You can filter the results of a ListContainerInstances operation with cluster query language statements. For more information, see Cluster Query Language in the Amazon Elastic Container Service Developer Guide.

status
Filters the container instances by status. For example, if you specify the DRAINING status, the results include only container instances that have been set to DRAINING using UpdateContainerInstancesState. If you do not specify this parameter, the default is to include container instances set to all states other than INACTIVE.

---

**Description**
Lists the services that are running in a specified cluster.

**Usage**
```r
ecs_list_services(
  maxResults = NULL,
  nextToken = NULL,
  cluster = NULL,
  launchType = NULL,
  schedulingStrategy = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```
Arguments

maxResults  Character. Pagination limit[optional]
nextToken   Characters. The token for the next page of results[optional]
cluster     Character. The short name or full Amazon Resource Name (ARN) of the cluster
            that hosts the services to list.
launchType  Character. The launch type for the services to list.
schedulingStrategy
            Character. The scheduling strategy for services to list.
simplify    Logical. Whether to simplify the result and handle nextToken in the response[optional]
others      Named list. The parameters that are not included in the function parameters and
            need to be added into the request[optional]
print_on_error  Logical. Whether to show an error message when a network error occurs.
retry_time   Integer. Number of retries for a REST request when encounter the network
            issue. If the request has been sent retry_time times but still not be able to get
            the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can
                   not be less than 1 ms.
region       Character. The region of the AWS service.

Value

A list object or a character vector

maxResults

Pagination limit

cluster

The short name or full Amazon Resource Name (ARN) of the cluster that hosts the services to list.
If you do not specify a cluster, the default cluster is assumed.

launchType

The launch type for the services to list.

schedulingStrategy

The scheduling strategy for services to list.
**Description**

List the tags for an Amazon ECS resource.

**Usage**

```r
ecs_list_tags_for_resource(
    resourceArn = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

**Arguments**

- **resourceArn** Character. The Amazon Resource Name (ARN) that identifies the resource for which to list the tags.
- **simplify** Logical. Whether to simplify the result and handle `nextToken` in the response[optional]
- **others** Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error** Logical. Whether to show an error message when a network error occurs.
- **retry_time** Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout** Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region** Character. The region of the AWS service.

**Value**

A list object or a character vector

**resourceArn**

The Amazon Resource Name (ARN) that identifies the resource for which to list the tags. Currently, the supported resources are Amazon ECS tasks, services, task definitions, clusters, and container instances.
Description

List Tasks

Usage

```r
ecs_list_tasks(
    maxResults = NULL,
    nextToken = NULL,
    cluster = NULL,
    containerInstance = NULL,
    family = NULL,
    startedBy = NULL,
    serviceName = NULL,
    desiredStatus = NULL,
    launchType = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

Arguments

maxResults  Character. Pagination limit[optional]
nextToken   Characters. The token for the next page of results[optional]
cluster     Character. The short name or full Amazon Resource Name (ARN) of the cluster that hosts the tasks to list.
containerInstance
            Character. The container instance ID or full ARN of the container instance with which to filter the ListTasks...
family      Character. The name of the family with which to filter the ListTasks results.
startedBy   Character. The startedBy value with which to filter the task results.
serviceName Character. The name of the service with which to filter the ListTasks results.
desiredStatus Character. The task desired status with which to filter the ListTasks results.
launchType  Character. The launch type for services to list.
simplify    Logical. Whether to simplify the result and handle nextToken in the response[optional]
others      Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error  Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region  Character. The region of the AWS service.

Value
A list object or a character vector

maxResults
Pagination limit

cluster
The short name or full Amazon Resource Name (ARN) of the cluster that hosts the tasks to list. If you do not specify a cluster, the default cluster is assumed.

containerInstance
The container instance ID or full ARN of the container instance with which to filter the ListTasks results. Specifying a containerInstance limits the results to tasks that belong to that container instance.

family
The name of the family with which to filter the ListTasks results. Specifying a family limits the results to tasks that belong to that family.

startedBy
The startedBy value with which to filter the task results. Specifying a startedBy value limits the results to tasks that were started with that value.

serviceName
The name of the service with which to filter the ListTasks results. Specifying a serviceName limits the results to tasks that belong to that service.

desiredStatus
The task desired status with which to filter the ListTasks results. Specifying a desiredStatus of STOPPED limits the results to tasks that Amazon ECS has set the desired status to STOPPED. This can be useful for debugging tasks that are not starting properly or have died or finished. The default status filter is RUNNING, which shows tasks that Amazon ECS has set the desired status to RUNNING.
Although you can filter results based on a desired status of PENDING, this does not return any results. Amazon ECS never sets the desired status of a task to that value (only a task's lastStatus may have a value of PENDING).

**launchType**

The launch type for services to list.

---

**Description**

Returns a list of task definitions that are registered to your account. You can filter the results by family name with the familyPrefix parameter or by status with the status parameter.

**Usage**

```r
ecs_list_task_definitions(
  maxResults = NULL,
  nextToken = NULL,
  familyPrefix = NULL,
  status = NULL,
  sort = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- **maxResults**: Character. Pagination limit[optional]
- **nextToken**: Characters. The token for the next page of results[optional]
- **familyPrefix**: Character. The full family name with which to filter the ListTaskDefinitions results.
- **status**: Character. The task definition status with which to filter the ListTaskDefinitions results.
- **sort**: Character. The order in which to sort the results.
- **simplify**: Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region Character. The region of the AWS service.

Value
A list object or a character vector

maxResults
Pagination limit

familyPrefix
The full family name with which to filter the ListTaskDefinitions results. Specifying a familyPrefix limits the listed task definitions to task definition revisions that belong to that family.

status
The task definition status with which to filter the ListTaskDefinitions results. By default, only ACTIVE task definitions are listed. By setting this parameter to INACTIVE, you can view task definitions that are INACTIVE as long as an active task or service still references them. If you paginate the resulting output, be sure to keep the status value constant in each subsequent request.

sort
The order in which to sort the results. Valid values are ASC and DESC. By default (ASC), task definitions are listed lexicographically by family name and in ascending numerical order by revision so that the newest task definitions in a family are listed last. Setting this parameter to DESC reverses the sort order on family name and revision so that the newest task definitions in a family are listed first.
Usage

```r
escs_list_task_definition_families(
  maxResults = NULL,
  nextToken = NULL,
  familyPrefix = NULL,
  status = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- `maxResults` Character. Pagination limit[optional]
- `nextToken` Characters. The token for the next page of results[optional]
- `familyPrefix` Character. The familyPrefix is a string that is used to filter the results of ListTaskDefinitionFamilies. If you specify a familyPrefix, only task definition family names that begin with the familyPrefix string are returned.
- `status` Character. The task definition family status with which to filter the ListTaskDefinitionFamilies results.
- `simplify` Logical. Whether to simplify the result and handle nextToken in the response[optional]
- `others` Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- `print_on_error` Logical. Whether to show an error message when a network error occurs.
- `retry_time` Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- `network_timeout` Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- `region` Character. The region of the AWS service.

Value

A list object or a character vector

**maxResults**

Pagination limit

**familyPrefix**

The familyPrefix is a string that is used to filter the results of ListTaskDefinitionFamilies. If you specify a familyPrefix, only task definition family names that begin with the familyPrefix string are returned.
status

The task definition family status with which to filter the ListTaskDefinitionFamilies results. By default, both ACTIVE and INACTIVE task definition families are listed. If this parameter is set to ACTIVE, only task definition families that have an ACTIVE task definition revision are returned. If this parameter is set to INACTIVE, only task definition families that do not have any ACTIVE task definition revisions are returned. If you paginate the resulting output, be sure to keep the status value constant in each subsequent request.

Description

Put Account Setting

Usage

ecs_put_account_setting(
  name = NULL,
  value = NULL,
  principalArn = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)

Arguments

- **name**: Character. The Amazon ECS resource name for which to modify the account setting.
- **value**: Character. The account setting value for the specified principal ARN.
- **principalArn**: Character. The ARN of the principal, which can be an IAM user, IAM role, or the root user.
- **simplify**: Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout

Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region

Character. The region of the AWS service.

Value

A list object or a character vector

name

The Amazon ECS resource name for which to modify the account setting. If serviceLongArnFormat is specified, the ARN for your Amazon ECS services is affected. If taskLongArnFormat is specified, the ARN and resource ID for your Amazon ECS tasks is affected. If containerInstanceLongArnFormat is specified, the ARN and resource ID for your Amazon ECS container instances is affected. If awsvpcTrunking is specified, the elastic network interface (ENI) limit for your Amazon ECS container instances is affected. If containerInsights is specified, the default setting for CloudWatch Container Insights for your clusters is affected.

value

The account setting value for the specified principal ARN. Accepted values are enabled and disabled.

principalArn

The ARN of the principal, which can be an IAM user, IAM role, or the root user. If you specify the root user, it modifies the account setting for all IAM users, IAM roles, and the root user of the account unless an IAM user or role explicitly overrides these settings. If this field is omitted, the setting is changed only for the authenticated user.

Description

Modifies an account setting for all IAM users on an account for whom no individual account setting has been specified. Account settings are set on a per-Region basis.

Usage

ecs_put_account_setting_default(
    name = NULL,
    value = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
)
network_timeout = aws_get_network_timeout()
region = aws_get_region()

**Arguments**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Character. The resource name for which to modify the account setting.</td>
</tr>
<tr>
<td>value</td>
<td>Character. The account setting value for the specified principal ARN.</td>
</tr>
<tr>
<td>simplify</td>
<td>Logical. Whether to simplify the result and handle nextToken in the response[optional]</td>
</tr>
<tr>
<td>others</td>
<td>Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]</td>
</tr>
<tr>
<td>print_on_error</td>
<td>Logical. Whether to show an error message when a network error occurs.</td>
</tr>
<tr>
<td>retry_time</td>
<td>Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.</td>
</tr>
<tr>
<td>network_timeout</td>
<td>Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.</td>
</tr>
<tr>
<td>region</td>
<td>Character. The region of the AWS service.</td>
</tr>
</tbody>
</table>

**Value**

A list object or a character vector

**name**

The resource name for which to modify the account setting. If serviceLongArnFormat is specified, the ARN for your Amazon ECS services is affected. If taskLongArnFormat is specified, the ARN and resource ID for your Amazon ECS tasks is affected. If containerInstanceLongArnFormat is specified, the ARN and resource ID for your Amazon ECS container instances is affected. If awsvpcTrunking is specified, the ENI limit for your Amazon ECS container instances is affected. If containerInsights is specified, the default setting for CloudWatch Container Insights for your clusters is affected.

**value**

The account setting value for the specified principal ARN. Accepted values are enabled and disabled.

---

**Description**

Create or update an attribute on an Amazon ECS resource. If the attribute does not exist, it is created. If the attribute exists, its value is replaced with the specified value. To delete an attribute, use DeleteAttributes. For more information, see Attributes in the Amazon Elastic Container Service Developer Guide.
Usage

escs_put_attributes(
    cluster = NULL,
    attributes = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

cluster Character. The short name or full Amazon Resource Name (ARN) of the cluster that contains the resource to apply...

attributes List. The attributes to apply to your resource.

simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]

others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.

Value

A list object or a character vector

cluster

The short name or full Amazon Resource Name (ARN) of the cluster that contains the resource to apply attributes. If you do not specify a cluster, the default cluster is assumed.

attributes

The attributes to apply to your resource. You can specify up to 10 custom attributes per resource. You can specify up to 10 attributes in a single call.
Description
Put Cluster Capacity Providers

Usage
ecs_put_cluster_capacity_providers(
    cluster = NULL,
    capacityProviders = NULL,
    defaultCapacityProviderStrategy = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

cluster       Character. The short name or full Amazon Resource Name (ARN) of the cluster
to modify the capacity provider...
capacityProviders List. The name of one or more capacity providers to associate with the cluster.
defaultCapacityProviderStrategy List. The capacity provider strategy to use by default for the cluster.
simplify      Logical. Whether to simplify the result and handle nextToken in the response[optional]
others        Named list. The parameters that are not included in the function parameters and
need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.
retry_time    Integer. Number of retries for a REST request when encounter the network
issue. If the request has been sent retry_time times but still not be able to get
the response, an error will be thrown.
network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can
not be less than 1 ms.
region        Character. The region of the AWS service.

Value
A list object or a character vector
cluster

The short name or full Amazon Resource Name (ARN) of the cluster to modify the capacity provider settings for. If you do not specify a cluster, the default cluster is assumed.

capacityProviders

The name of one or more capacity providers to associate with the cluster.
If specifying a capacity provider that uses an Auto Scaling group, the capacity provider must already be created. New capacity providers can be created with the CreateCapacityProvider API operation.
To use a AWS Fargate capacity provider, specify either the FARGATE or FARGATE_SPOT capacity providers. The AWS Fargate capacity providers are available to all accounts and only need to be associated with a cluster to be used.

defaultCapacityProviderStrategy

The capacity provider strategy to use by default for the cluster.
When creating a service or running a task on a cluster, if no capacity provider or launch type is specified then the default capacity provider strategy for the cluster is used.
A capacity provider strategy consists of one or more capacity providers along with the base and weight to assign to them. A capacity provider must be associated with the cluster to be used in a capacity provider strategy. The PutClusterCapacityProviders API is used to associate a capacity provider with a cluster. Only capacity providers with an ACTIVE or UPDATING status can be used.
If specifying a capacity provider that uses an Auto Scaling group, the capacity provider must already be created. New capacity providers can be created with the CreateCapacityProvider API operation.
To use a AWS Fargate capacity provider, specify either the FARGATE or FARGATE_SPOT capacity providers. The AWS Fargate capacity providers are available to all accounts and only need to be associated with a cluster to be used.

Description

Register Container Instance

Usage

```
ecs_register_container_instance(
    cluster = NULL,
    instanceIdentityDocument = NULL,
    instanceIdentityDocumentSignature = NULL,
    totalResources = NULL,
    versionInfo = NULL,
    containerInstanceArn = NULL,
```

ecs_register_container_instance

attributes = NULL,
platformDevices = NULL,
tags = NULL,
simplify = TRUE,
others = list(),
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

cluster
Character. The short name or full Amazon Resource Name (ARN) of the cluster with which to register your container...

instanceIdentityDocument
Character. The instance identity document for the EC2 instance to register.

instanceIdentityDocumentSignature
Character. The instance identity document signature for the EC2 instance to register.

totalResources
List. The resources available on the instance.

versionInfo
Object. The version information for the Amazon ECS container agent and Docker daemon running on the container...

containerInstanceArn
Character. The ARN of the container instance (if it was previously registered).

attributes
List. The container instance attributes that this container instance supports.

platformDevices
List. The devices that are available on the container instance. The only supported device type is a GPU.

tags
List. The metadata that you apply to the container instance to help you categorize and organize them.

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

Value

A list object or a character vector
**cluster**

The short name or full Amazon Resource Name (ARN) of the cluster with which to register your container instance. If you do not specify a cluster, the default cluster is assumed.

**instanceIdentityDocument**

The instance identity document for the EC2 instance to register. This document can be found by running the following command from the instance: `curl http://169.254.169.254/latest/dynamic/instance-identity/document/`

**instanceIdentityDocumentSignature**

The instance identity document signature for the EC2 instance to register. This signature can be found by running the following command from the instance: `curl http://169.254.169.254/latest/dynamic/instance-identity/signature/`

**totalResources**

The resources available on the instance.

**versionInfo**

The version information for the Amazon ECS container agent and Docker daemon running on the container instance.

**containerInstanceArn**

The ARN of the container instance (if it was previously registered).

**attributes**

The container instance attributes that this container instance supports.

**platformDevices**

The devices that are available on the container instance. The only supported device type is a GPU.

**tags**

The metadata that you apply to the container instance to help you categorize and organize them. Each tag consists of a key and an optional value, both of which you define.

The following basic restrictions apply to tags:

- Maximum number of tags per resource - 50
- For each resource, each tag key must be unique, and each tag key can have only one value.
- Maximum key length - 128 Unicode characters in UTF-8
- Maximum value length - 256 Unicode characters in UTF-8
- If your tagging schema is used across multiple services and resources, remember that other services may have restrictions on allowed characters. Generally allowed characters are: letters, numbers, and spaces representable in UTF-8, and the following characters: + - . _ : / @.
• Tag keys and values are case-sensitive.
• Do not use `aws:`, `AWS:`, or any upper or lowercase combination of such as a prefix for either keys or values as it is reserved for AWS use. You cannot edit or delete tag keys or values with this prefix. Tags with this prefix do not count against your tags per resource limit.

---

**ecs_register_task_definition**

*Register Task Definition*

**Description**

Register Task Definition

**Usage**

```r
ecs_register_task_definition(
    family = NULL,
    taskRoleArn = NULL,
    executionRoleArn = NULL,
    networkMode = NULL,
    containerDefinitions = NULL,
    volumes = NULL,
    placementConstraints = NULL,
    requiresCompatibilities = NULL,
    cpu = NULL,
    memory = NULL,
    tags = NULL,
    pidMode = NULL,
    ipcMode = NULL,
    proxyConfiguration = NULL,
    inferenceAccelerators = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

**Arguments**

- **family**  
  Character. You must specify a `family` for a task definition, which allows you to track multiple versions of...

- **taskRoleArn**  
  Character. The short name or full Amazon Resource Name (ARN) of the IAM role that contains in this task can...
executionRoleArn
Character. The Amazon Resource Name (ARN) of the task execution role that grants the Amazon ECS container agent...

networkMode
Character. The Docker networking mode to use for the containers in the task.

containerDefinitions
List. A list of container definitions in JSON format that describe the different containers that make up...

volumes
List. A list of volume definitions in JSON format that containers in your task may use.

placementConstraints
List. An array of placement constraint objects to use for the task.

requiresCompatibilities
List. The task launch type that Amazon ECS should validate the task definition against.

cpu
Character. The number of CPU units used by the task.

memory
Character. The amount of memory (in MiB) used by the task.

tags
List. The metadata that you apply to the task definition to help you categorize and organize them.

pidMode
Character. The process namespace to use for the containers in the task.

ipcMode
Character. The IPC resource namespace to use for the containers in the task.

proxyConfiguration
No description can be found.

inferenceAccelerators
List. The Elastic Inference accelerators to use for the containers in the task.

simplify
Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
Logical. Whether to show an error message when a network error occurs.

retry_time
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
Character. The region of the AWS service.

**Value**
A list object or a character vector

**family**
You must specify a family for a task definition, which allows you to track multiple versions of the same task definition. The family is used as a name for your task definition. Up to 255 letters (uppercase and lowercase), numbers, and hyphens are allowed.
**ecs_register_task_definition**

**taskRoleArn**

The short name or full Amazon Resource Name (ARN) of the IAM role that containers in this task can assume. All containers in this task are granted the permissions that are specified in this role. For more information, see IAM Roles for Tasks in the Amazon Elastic Container Service Developer Guide.

**executionRoleArn**

The Amazon Resource Name (ARN) of the task execution role that grants the Amazon ECS container agent permission to make AWS API calls on your behalf. The task execution IAM role is required depending on the requirements of your task. For more information, see Amazon ECS task execution IAM role in the Amazon Elastic Container Service Developer Guide.

**networkMode**

The Docker networking mode to use for the containers in the task. The valid values are none, bridge, awsvpc, and host. If no network mode is specified, the default is bridge.

For Amazon ECS tasks on Fargate, the awsvpc network mode is required. For Amazon ECS tasks on Amazon EC2 instances, any network mode can be used. If the network mode is set to none, you cannot specify port mappings in your container definitions, and the tasks containers do not have external connectivity. The host and awsvpc network modes offer the highest networking performance for containers because they use the EC2 network stack instead of the virtualized network stack provided by the bridge mode.

With the host and awsvpc network modes, exposed container ports are mapped directly to the corresponding host port (for the host network mode) or the attached elastic network interface port (for the awsvpc network mode), so you cannot take advantage of dynamic host port mappings.

When using the host network mode, you should not run containers using the root user (UID 0). It is considered best practice to use a non-root user.

If the network mode is awsvpc, the task is allocated an elastic network interface, and you must specify a NetworkConfiguration value when you create a service or run a task with the task definition. For more information, see Task Networking in the Amazon Elastic Container Service Developer Guide.

Currently, only Amazon ECS-optimized AMIs, other Amazon Linux variants with the ecs-init package, or AWS Fargate infrastructure support the awsvpc network mode.

If the network mode is host, you cannot run multiple instantiations of the same task on a single container instance when port mappings are used.

Docker for Windows uses different network modes than Docker for Linux. When you register a task definition with Windows containers, you must not specify a network mode. If you use the console to register a task definition with Windows containers, you must choose the “network mode object. For more information, see Network settings in the Docker run reference.

**containerDefinitions**

A list of container definitions in JSON format that describe the different containers that make up your task.
volumes

A list of volume definitions in JSON format that containers in your task may use.

placementConstraints

An array of placement constraint objects to use for the task. You can specify a maximum of 10 constraints per task (this limit includes constraints in the task definition and those specified at runtime).

requiresCompatibilities

The task launch type that Amazon ECS should validate the task definition against. This ensures that the task definition parameters are compatible with the specified launch type. If no value is specified, it defaults to EC2.

cpu

The number of CPU units used by the task. It can be expressed as an integer using CPU units, for example 1024, or as a string using vCPUs, for example 1 vCPU or 1 vcpu, in a task definition. String values are converted to an integer indicating the CPU units when the task definition is registered.

Task-level CPU and memory parameters are ignored for Windows containers. We recommend specifying container-level resources for Windows containers.

If you are using the EC2 launch type, this field is optional. Supported values are between 128 CPU units (0.125 vCPUs) and 10240 CPU units (10 vCPUs).

If you are using the Fargate launch type, this field is required and you must use one of the following values, which determines your range of supported values for the memory parameter:

- 256 (.25 vCPU) - Available memory values: 512 (0.5 GB), 1024 (1 GB), 2048 (2 GB)
- 512 (.5 vCPU) - Available memory values: 1024 (1 GB), 2048 (2 GB), 3072 (3 GB), 4096 (4 GB)
- 1024 (1 vCPU) - Available memory values: 2048 (2 GB), 3072 (3 GB), 4096 (4 GB), 5120 (5 GB), 6144 (6 GB), 7168 (7 GB), 8192 (8 GB)
- 2048 (2 vCPU) - Available memory values: Between 4096 (4 GB) and 16384 (16 GB) in increments of 1024 (1 GB)
- 4096 (4 vCPU) - Available memory values: Between 8192 (8 GB) and 30720 (30 GB) in increments of 1024 (1 GB)

memory

The amount of memory (in MiB) used by the task. It can be expressed as an integer using MiB, for example 1024, or as a string using GB, for example 1GB or 1 GB, in a task definition. String values are converted to an integer indicating the MiB when the task definition is registered.

Task-level CPU and memory parameters are ignored for Windows containers. We recommend specifying container-level resources for Windows containers.

If using the EC2 launch type, this field is optional.

If using the Fargate launch type, this field is required and you must use one of the following values, which determines your range of supported values for the cpu parameter:
ecs_register_task_definition

- 512 (0.5 GB), 1024 (1 GB), 2048 (2 GB) - Available cpu values: 256 (.25 vCPU)
- 1024 (1 GB), 2048 (2 GB), 3072 (3 GB), 4096 (4 GB) - Available cpu values: 512 (.5 vCPU)
- 2048 (2 GB), 3072 (3 GB), 4096 (4 GB), 5120 (5 GB), 6144 (6 GB), 7168 (7 GB), 8192 (8 GB) - Available cpu values: 1024 (1 vCPU)
- Between 4096 (4 GB) and 16384 (16 GB) in increments of 1024 (1 GB) - Available cpu values: 2048 (2 vCPU)
- Between 8192 (8 GB) and 30720 (30 GB) in increments of 1024 (1 GB) - Available cpu values: 4096 (4 vCPU)

tags

The metadata that you apply to the task definition to help you categorize and organize them. Each tag consists of a key and an optional value, both of which you define.

The following basic restrictions apply to tags:

- Maximum number of tags per resource - 50
- For each resource, each tag key must be unique, and each tag key can have only one value.
- Maximum key length - 128 Unicode characters in UTF-8
- Maximum value length - 256 Unicode characters in UTF-8
- If your tagging schema is used across multiple services and resources, remember that other services may have restrictions on allowed characters. Generally allowed characters are: letters, numbers, and spaces representable in UTF-8, and the following characters: + - = . _ : / @.
- Tag keys and values are case-sensitive.
- Do not use aws:, AWS:, or any upper or lowercase combination of such as a prefix for either keys or values as it is reserved for AWS use. You cannot edit or delete tag keys or values with this prefix. Tags with this prefix do not count against your tags per resource limit.

pidMode

The process namespace to use for the containers in the task. The valid values are host or task. If host is specified, then all containers within the tasks that specified the host PID mode on the same container instance share the same process namespace with the host Amazon EC2 instance. If task is specified, all containers within the specified task share the same process namespace. If no value is specified, the default is a private namespace. For more information, see PID settings in the Docker run reference.

If the host PID mode is used, be aware that there is a heightened risk of undesired process namespace expose. For more information, see Docker security.

This parameter is not supported for Windows containers or tasks using the Fargate launch type.

ipcMode

The IPC resource namespace to use for the containers in the task. The valid values are host, task, or none. If host is specified, then all containers within the tasks that specified the host IPC mode on the same container instance share the same IPC resources with the host Amazon EC2 instance. If task is specified, all containers within the specified task share the same IPC resources. If none
is specified, then IPC resources within the containers of a task are private and not shared with other containers in a task or on the container instance. If no value is specified, then the IPC resource namespace sharing depends on the Docker daemon setting on the container instance. For more information, see IPC settings in the Docker run reference.

If the host IPC mode is used, be aware that there is a heightened risk of undesired IPC namespace expose. For more information, see Docker security.

If you are setting namespaced kernel parameters using systemControls for the containers in the task, the following will apply to your IPC resource namespace. For more information, see System Controls in the Amazon Elastic Container Service Developer Guide.

- For tasks that use the host IPC mode, IPC namespace related systemControls are not supported.
- For tasks that use the task IPC mode, IPC namespace related systemControls will apply to all containers within a task.

This parameter is not supported for Windows containers or tasks using the Fargate launch type.

**proxyConfiguration**

No description can be found.

**inferenceAccelerators**

The Elastic Inference accelerators to use for the containers in the task.

---

### ecs_run_task

**Run Task**

#### Description

Run Task

#### Usage

```python
capacityProviderStrategy = NULL,
cluster = NULL,
count = NULL,
enableECSManagedTags = NULL,
group = NULL,
launchType = NULL,
networkConfiguration = NULL,
overrides = NULL,
placementConstraints = NULL,
placementStrategy = NULL,
platformVersion = NULL,
propagateTags = NULL,
```
referenceId = NULL,
startedBy = NULL,
tags = NULL,
taskDefinition = NULL,
simplify = TRUE,
others = list(),
print_on_error = aws_get_print_on_error(),
retry_time = aws_get_retry_time(),
network_timeout = aws_get_network_timeout(),
region = aws_get_region()
)

Arguments

capacityProviderStrategy List. The capacity provider strategy to use for the task.

cluster Character. The short name or full Amazon Resource Name (ARN) of the cluster on which to run your task.

count Integer. The number of instantiations of the specified task to place on your cluster.

enableECSManagedTags Logical. Specifies whether to enable Amazon ECS managed tags for the task.

group Character. The name of the task group to associate with the task.

launchType Character. The launch type on which to run your task.

networkConfiguration Object. The network configuration for the task.

overrides Object. A list of container overrides in JSON format that specify the name of a container in the specified...

placementConstraints List. An array of placement constraint objects to use for the task.

placementStrategy List. The placement strategy objects to use for the task.

platformVersion Character. The platform version the task should run.

propagateTags Character. Specifies whether to propagate the tags from the task definition to the task.

referenceId Character. The reference ID to use for the task.

startedBy Character. An optional tag specified when a task is started.

tags List. The metadata that you apply to the task to help you categorize and organize them.

taskDefinition Character. The family and revision (family:revision) or full ARN of the task definition to run.

simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]

others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.

Value
A list object or a character vector

capacityProviderStrategy
The capacity provider strategy to use for the task.
A capacity provider strategy consists of one or more capacity providers along with the base and weight to assign to them. A capacity provider must be associated with the cluster to be used in a capacity provider strategy. The PutClusterCapacityProviders API is used to associate a capacity provider with a cluster. Only capacity providers with an ACTIVE or UPDATING status can be used.
If a capacityProviderStrategy is specified, the launchType parameter must be omitted. If no capacityProviderStrategy or launchType is specified, the defaultCapacityProviderStrategy for the cluster is used.
If specifying a capacity provider that uses an Auto Scaling group, the capacity provider must already be created. New capacity providers can be created with the CreateCapacityProvider API operation.
To use a AWS Fargate capacity provider, specify either the FARGATE or FARGATE_SPOT capacity providers. The AWS Fargate capacity providers are available to all accounts and only need to be associated with a cluster to be used.
The PutClusterCapacityProviders API operation is used to update the list of available capacity providers for a cluster after the cluster is created.

count
The number of instantiations of the specified task to place on your cluster. You can specify up to 10 tasks per call.

enableECSManagedTags
Specifies whether to enable Amazon ECS managed tags for the task. For more information, see Tagging Your Amazon ECS Resources in the Amazon Elastic Container Service Developer Guide.
**group**

The name of the task group to associate with the task. The default value is the family name of the task definition (for example, family:my-family-name).

**launchType**

The launch type on which to run your task. For more information, see Amazon ECS Launch Types in the Amazon Elastic Container Service Developer Guide.

If a launchType is specified, the capacityProviderStrategy parameter must be omitted.

**networkConfiguration**

The network configuration for the task. This parameter is required for task definitions that use the awsvpc network mode to receive their own elastic network interface, and it is not supported for other network modes. For more information, see Task Networking in the Amazon Elastic Container Service Developer Guide.

**overrides**

A list of container overrides in JSON format that specify the name of a container in the specified task definition and the overrides it should receive. You can override the default command for a container (that is specified in the task definition or Docker image) with a command override. You can also override existing environment variables (that are specified in the task definition or Docker image) on a container or add new environment variables to it with an environment override.

A total of 8192 characters are allowed for overrides. This limit includes the JSON formatting characters of the override structure.

**placementConstraints**

An array of placement constraint objects to use for the task. You can specify up to 10 constraints per task (including constraints in the task definition and those specified at runtime).

**placementStrategy**

The placement strategy objects to use for the task. You can specify a maximum of five strategy rules per task.

**platformVersion**

The platform version the task should run. A platform version is only specified for tasks using the Fargate launch type. If one is not specified, the LATEST platform version is used by default. For more information, see AWS Fargate Platform Versions in the Amazon Elastic Container Service Developer Guide.
propagateTags

Specifies whether to propagate the tags from the task definition to the task. If no value is specified, the tags are not propagated. Tags can only be propagated to the task during task creation. To add tags to a task after task creation, use the TagResource API action.

An error will be received if you specify the SERVICE option when running a task.

referenceId

The reference ID to use for the task.

startedBy

An optional tag specified when a task is started. For example, if you automatically trigger a task to run a batch process job, you could apply a unique identifier for that job to your task with the startedBy parameter. You can then identify which tasks belong to that job by filtering the results of a ListTasks call with the startedBy value. Up to 36 letters (uppercase and lowercase), numbers, hyphens, and underscores are allowed.

If a task is started by an Amazon ECS service, then the startedBy parameter contains the deployment ID of the service that starts it.

tags

The metadata that you apply to the task to help you categorize and organize them. Each tag consists of a key and an optional value, both of which you define.

The following basic restrictions apply to tags:

- Maximum number of tags per resource - 50
- For each resource, each tag key must be unique, and each tag key can have only one value.
- Maximum key length - 128 Unicode characters in UTF-8
- Maximum value length - 256 Unicode characters in UTF-8
- If your tagging schema is used across multiple services and resources, remember that other services may have restrictions on allowed characters. Generally allowed characters are: letters, numbers, and spaces representable in UTF-8, and the following characters: + = . _ : / @.
- Tag keys and values are case-sensitive.
- Do not use aws:, AWS:, or any upper or lowercase combination of such as a prefix for either keys or values as it is reserved for AWS use. You cannot edit or delete tag keys or values with this prefix. Tags with this prefix do not count against your tags per resource limit.

taskDefinition

The family and revision (family:revision) or full ARN of the task definition to run. If a revision is not specified, the latest ACTIVE revision is used.
Description

Start Task

Usage

```r
ecs_start_task(
  cluster = NULL,
  containerInstances = NULL,
  enableECSManagedTags = NULL,
  group = NULL,
  networkConfiguration = NULL,
  overrides = NULL,
  propagateTags = NULL,
  referenceId = NULL,
  startedBy = NULL,
  tags = NULL,
  taskDefinition = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **cluster**: Character. The short name or full Amazon Resource Name (ARN) of the cluster on which to start your task.
- **containerInstances**: List. The container instance IDs or full ARN entries for the container instances on which you would like...
- **enableECSManagedTags**: Logical. Specifies whether to enable Amazon ECS managed tags for the task.
- **group**: Character. The name of the task group to associate with the task.
- **networkConfiguration**: Object. The VPC subnet and security group configuration for tasks that receive their own elastic network...
- **overrides**: Object. A list of container overrides in JSON format that specify the name of a container in the specified...
- **propagateTags**: Character. Specifies whether to propagate the tags from the task definition or the service to the task.
referenceId  Character. The reference ID to use for the task.

startedBy  Character. An optional tag specified when a task is started.

tags  List. The metadata that you apply to the task to help you categorize and organize them.

taskDefinition  Character. The family and revision (family:revision) or full ARN of the task definition to start.

simplify  Logical. Whether to simplify the result and handle nextToken in the response[optional]

others  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error  Logical. Whether to show an error message when a network error occurs.

retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  Character. The region of the AWS service.

Value

A list object or a character vector

cluster

The short name or full Amazon Resource Name (ARN) of the cluster on which to start your task. If you do not specify a cluster, the default cluster is assumed.

containerInstances

The container instance IDs or full ARN entries for the container instances on which you would like to place your task. You can specify up to 10 container instances.

enableECSManagedTags

Specifies whether to enable Amazon ECS managed tags for the task. For more information, see Tagging Your Amazon ECS Resources in the Amazon Elastic Container Service Developer Guide.

group

The name of the task group to associate with the task. The default value is the family name of the task definition (for example, family:my-family-name).

networkConfiguration

The VPC subnet and security group configuration for tasks that receive their own elastic network interface by using the awsvpc networking mode.
overrides

A list of container overrides in JSON format that specify the name of a container in the specified task definition and the overrides it should receive. You can override the default command for a container (that is specified in the task definition or Docker image) with a command override. You can also override existing environment variables (that are specified in the task definition or Docker image) on a container or add new environment variables to it with an environment override.

A total of 8192 characters are allowed for overrides. This limit includes the JSON formatting characters of the override structure.

propagateTags

Specifies whether to propagate the tags from the task definition or the service to the task. If no value is specified, the tags are not propagated.

referenceId

The reference ID to use for the task.

startedBy

An optional tag specified when a task is started. For example, if you automatically trigger a task to run a batch process job, you could apply a unique identifier for that job to your task with the startedBy parameter. You can then identify which tasks belong to that job by filtering the results of a ListTasks call with the startedBy value. Up to 36 letters (uppercase and lowercase), numbers, hyphens, and underscores are allowed.

If a task is started by an Amazon ECS service, then the startedBy parameter contains the deployment ID of the service that starts it.

tags

The metadata that you apply to the task to help you categorize and organize them. Each tag consists of a key and an optional value, both of which you define.

The following basic restrictions apply to tags:

- Maximum number of tags per resource - 50
- For each resource, each tag key must be unique, and each tag key can have only one value.
- Maximum key length - 128 Unicode characters in UTF-8
- Maximum value length - 256 Unicode characters in UTF-8
- If your tagging schema is used across multiple services and resources, remember that other services may have restrictions on allowed characters. Generally allowed characters are: letters, numbers, and spaces representable in UTF-8, and the following characters: + - _ : / @.
- Tag keys and values are case-sensitive.
- Do not use aws:, AWS:, or any upper or lowercase combination of such as a prefix for either keys or values as it is reserved for AWS use. You cannot edit or delete tag keys or values with this prefix. Tags with this prefix do not count against your tags per resource limit.
taskDefinition

The family and revision (family:revision) or full ARN of the task definition to start. If a revision is not specified, the latest ACTIVE revision is used.

--------

taskDefinition

ecs_stop_task

Description

Stop Task

Usage

ecs_stop_task(
    cluster = NULL,
    task = NULL,
    reason = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

Arguments

cluster Character. The short name or full Amazon Resource Name (ARN) of the cluster that hosts the task to stop.

task Character. The task ID or full Amazon Resource Name (ARN) of the task to stop.

reason Character. An optional message specified when a task is stopped.

simplify Logical. Whether to simplify the result and handle nextToken in the response[optional]

others Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error Logical. Whether to show an error message when a network error occurs.

retry_time Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region Character. The region of the AWS service.
Value
A list object or a character vector

cluster
The short name or full Amazon Resource Name (ARN) of the cluster that hosts the task to stop. If you do not specify a cluster, the default cluster is assumed.

task
The task ID or full Amazon Resource Name (ARN) of the task to stop.

reason
An optional message specified when a task is stopped. For example, if you are using a custom scheduler, you can use this parameter to specify the reason for stopping the task here, and the message appears in subsequent DescribeTasks API operations on this task. Up to 255 characters are allowed in this message.

### Usage

ecs_submit_attachment_state_changes(
    cluster = NULL,
    attachments = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)

### Arguments

- **cluster**  Character. The short name or full ARN of the cluster that hosts the container instance the attachment belongs...
- **attachments**  List. Any attachments associated with the state change request.
- **simplify**  Logical. Whether to simplify the result and handle nextToken in the response[optional]
ecs_submit_container_state_change

**Description**
Submit Container State Change

**Usage**
```r
ces_submit_container_state_change(
    cluster = NULL,
    task = NULL,
    containerName = NULL,
    runtimeId = NULL,
    status = NULL,
    exitCode = NULL,
    reason = NULL,
    networkBindings = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error()
)```

**Value**
A list object or a character vector

**cluster**
The short name or full ARN of the cluster that hosts the container instance the attachment belongs to.

**attachments**
Any attachments associated with the state change request.

**others**
Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]

**print_on_error**
Logical. Whether to show an error message when a network error occurs.

**retry_time**
Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

**network_timeout**
Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

**region**
Character. The region of the AWS service.
Argument parameters

cluster  Character. The short name or full ARN of the cluster that hosts the container.

task  Character. The task ID or full Amazon Resource Name (ARN) of the task that hosts the container.

containerName  Character. The name of the container.

runtimeId  Character. The ID of the Docker container.

status  Character. The status of the state change request.

exitCode  Integer. The exit code returned for the state change request.

reason  Character. The reason for the state change request.

networkBindings  List. The network bindings of the container.

simplify  Logical. Whether to simplify the result and handle nextToken in the response[optional]

others  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error  Logical. Whether to show an error message when a network error occurs.

retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  Character. The region of the AWS service.

Value

A list object or a character vector

cluster

The short name or full ARN of the cluster that hosts the container.

task

The task ID or full Amazon Resource Name (ARN) of the task that hosts the container.

containerName

The name of the container.
ecs_submit_task_state_change

runtimeId
The ID of the Docker container.

status
The status of the state change request.

exitCode
The exit code returned for the state change request.

reason
The reason for the state change request.

networkBindings
The network bindings of the container.

ecs_submit_task_state_change

Submit Task State Change

Description
Submit Task State Change

Usage
ecs_submit_task_state_change(
    cluster = NULL,
    task = NULL,
    status = NULL,
    reason = NULL,
    containers = NULL,
    attachments = NULL,
    pullStartedAt = NULL,
    pullStoppedAt = NULL,
    executionStoppedAt = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
Arguments

cluster  Character. The short name or full Amazon Resource Name (ARN) of the cluster that hosts the task.

task  Character. The task ID or full ARN of the task in the state change request.

status  Character. The status of the state change request.

reason  Character. The reason for the state change request.

containers  List. Any containers associated with the state change request.

attachments  List. Any attachments associated with the state change request.

pullStartedAt  Character. The Unix timestamp for when the container image pull began.

pullStoppedAt  Character. The Unix timestamp for when the container image pull completed.

executionStoppedAt  Character. The Unix timestamp for when the task execution stopped.

simplify  Logical. Whether to simplify the result and handle nextToken in the response[optional]

others  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error  Logical. Whether to show an error message when a network error occurs.

retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  Character. The region of the AWS service.

Value

A list object or a character vector

cluster

The short name or full Amazon Resource Name (ARN) of the cluster that hosts the task.

task

The task ID or full ARN of the task in the state change request.

status

The status of the state change request.

reason

The reason for the state change request.
containers

Any containers associated with the state change request.

attachments

Any attachments associated with the state change request.

pullStartedAt

The Unix timestamp for when the container image pull began.

pullStoppedAt

The Unix timestamp for when the container image pull completed.

executionStoppedAt

The Unix timestamp for when the task execution stopped.

---

tag_resource

Tag Resource

Description

Associates the specified tags to a resource with the specified resourceArn. If existing tags on a resource are not specified in the request parameters, they are not changed. When a resource is deleted, the tags associated with that resource are deleted as well.

Usage

```r
ecs_tag_resource(
    resourceArn = NULL,
    tags = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```
Arguments

resourceArn  Character. The Amazon Resource Name (ARN) of the resource to which to add tags.
tags  List. The tags to add to the resource.
simplify  Logical. Whether to simplify the result and handle nextToken in the response[optional]
others  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
print_on_error  Logical. Whether to show an error message when a network error occurs.
retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
region  Character. The region of the AWS service.

Value

A list object or a character vector

resourceArn

The Amazon Resource Name (ARN) of the resource to which to add tags. Currently, the supported resources are Amazon ECS capacity providers, tasks, services, task definitions, clusters, and container instances.

tags

The tags to add to the resource. A tag is an array of key-value pairs.
The following basic restrictions apply to tags:

* Maximum number of tags per resource - 50
* For each resource, each tag key must be unique, and each tag key can have only one value.
* Maximum key length - 128 Unicode characters in UTF-8
* Maximum value length - 256 Unicode characters in UTF-8
* If your tagging schema is used across multiple services and resources, remember that other services may have restrictions on allowed characters. Generally allowed characters are: letters, numbers, and spaces representable in UTF-8, and the following characters: + - = . _ : / @.
* Tag keys and values are case-sensitive.
* Do not use aws:, AWS:, or any upper or lowercase combination of such as a prefix for either keys or values as it is reserved for AWS use. You cannot edit or delete tag keys or values with this prefix. Tags with this prefix do not count against your tags per resource limit.
**Description**

Deletes specified tags from a resource.

**Usage**

```r
ecs_untag_resource(
  resourceArn = NULL,
  tagKeys = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- `resourceArn` Character. The Amazon Resource Name (ARN) of the resource from which to delete tags.
- `tagKeys` List. The keys of the tags to be removed.
- `simplify` Logical. Whether to simplify the result and handle `nextToken` in the response[optional]
- `others` Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- `print_on_error` Logical. Whether to show an error message when a network error occurs.
- `retry_time` Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- `network_timeout` Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- `region` Character. The region of the AWS service.

**Value**

A list object or a character vector

**resourceArn**

The Amazon Resource Name (ARN) of the resource from which to delete tags. Currently, the supported resources are Amazon ECS capacity providers, tasks, services, task definitions, clusters, and container instances.
**tagKeys**

The keys of the tags to be removed.

---

**ecs_update_capacity_provider**

*Update Capacity Provider*

---

**Description**

Modifies the parameters for a capacity provider.

**Usage**

```r
ecs_update_capacity_provider(
  name = NULL,
  autoScalingGroupProvider = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- **name**
  - Character. The name of the capacity provider to update.

- **autoScalingGroupProvider**
  - Object. An object representing the parameters to update for the Auto Scaling group capacity provider.

- **simplify**
  - Logical. Whether to simplify the result and handle `nextToken` in the response [optional]

- **others**
  - Named list. The parameters that are not included in the function parameters and need to be added into the request [optional]

- **print_on_error**
  - Logical. Whether to show an error message when a network error occurs.

- **retry_time**
  - Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.

- **network_timeout**
  - Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

- **region**
  - Character. The region of the AWS service.

**Value**

A list object or a character vector
**name**

The name of the capacity provider to update.

**autoScalingGroupProvider**

An object representing the parameters to update for the Auto Scaling group capacity provider.

---

**ecs_update_cluster_settings**

*Update Cluster Settings*

---

**Description**

Modifies the settings to use for a cluster.

**Usage**

```r
ecs_update_cluster_settings(
    cluster = NULL,
    settings = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```

**Arguments**

- **cluster** Character. The name of the cluster to modify the settings for.
- **settings** List. The setting to use by default for a cluster.
- **simplify** Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others** Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error** Logical. Whether to show an error message when a network error occurs.
- **retry_time** Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- **network_timeout** Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region** Character. The region of the AWS service.
Value

A list object or a character vector

cluster

The name of the cluster to modify the settings for.

settings

The setting to use by default for a cluster. This parameter is used to enable CloudWatch Container Insights for a cluster. If this value is specified, it will override the containerInsights value set with PutAccountSetting or PutAccountSettingDefault.

Description

Update Container Agent

Usage

```r
ecs_update_container_agent(
  cluster = NULL,
  containerInstance = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

Arguments

- **cluster**: Character. The short name or full Amazon Resource Name (ARN) of the cluster that your container instance is...
- **containerInstance**: Character. The container instance ID or full ARN entries for the container instance on which you would like...
- **simplify**: Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
**Description**

Update Container Instances State

**Usage**

```r
esc_update_container_instances_state(
    cluster = NULL,
    containerInstances = NULL,
    status = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```
Arguments

cluster  Character. The short name or full Amazon Resource Name (ARN) of the cluster that hosts the container instance...

containerInstances  List. A list of container instance IDs or full ARN entries.

status  Character. The container instance state with which to update the container instance.

simplify  Logical. Whether to simplify the result and handle nextToken in the response[optional]

others  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error  Logical. Whether to show an error message when a network error occurs.

retry_time  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region  Character. The region of the AWS service.

Value

A list object or a character vector

cluster

The short name or full Amazon Resource Name (ARN) of the cluster that hosts the container instance to update. If you do not specify a cluster, the default cluster is assumed.

containerInstances

A list of container instance IDs or full ARN entries.

status

The container instance state with which to update the container instance. The only valid values for this action are ACTIVE and DRAINING. A container instance can only be updated to DRAINING status once it has reached an ACTIVE state. If a container instance is in REGISTERING, Deregistering, or REGISTRATION_FAILED state you can describe the container instance but will be unable to update the container instance state.
**ecs_update_service**  
*Update Service*

**Description**
Update Service

**Usage**

```r
ecs_update_service(
  cluster = NULL,
  service = NULL,
  desiredCount = NULL,
  taskDefinition = NULL,
  capacityProviderStrategy = NULL,
  deploymentConfiguration = NULL,
  networkConfiguration = NULL,
  placementConstraints = NULL,
  placementStrategy = NULL,
  platformVersion = NULL,
  forceNewDeployment = NULL,
  healthCheckGracePeriodSeconds = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- `cluster`  
  Character. The short name or full Amazon Resource Name (ARN) of the cluster that your service is running on.

- `service`  
  Character. The name of the service to update.

- `desiredCount`  
  Integer. The number of instantiations of the task to place and keep running in your service.

- `taskDefinition`  
  Character. The family and revision (family:revision) or full ARN of the task definition to run in your service.

- `capacityProviderStrategy`  
  List. The capacity provider strategy to update the service to use.

- `deploymentConfiguration`  
  Object. Optional deployment parameters that control how many tasks run during the deployment and the ordering...

- `networkConfiguration`  
  No description can be found.
placementConstraints
  List. An array of task placement constraint objects to update the service to use.

placementStrategy
  List. The task placement strategy objects to update the service to use.

platformVersion
  Character. The platform version on which your tasks in the service are running.

forceNewDeployment
  Logical. Whether to force a new deployment of the service.

healthCheckGracePeriodSeconds
  Integer. The period of time, in seconds, that the Amazon ECS service scheduler should ignore unhealthy Elastic...

simplify
  Logical. Whether to simplify the result and handle nextToken in the response[optional]

others
  Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]

print_on_error
  Logical. Whether to show an error message when a network error occurs.

retry_time
  Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.

network_timeout
  Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.

region
  Character. The region of the AWS service.

Value
  A list object or a character vector

cluster
  The short name or full Amazon Resource Name (ARN) of the cluster that your service is running on. If you do not specify a cluster, the default cluster is assumed.

service
  The name of the service to update.

desiredCount
  The number of instantiations of the task to place and keep running in your service.

taskDefinition
  The family and revision (family:revision) or full ARN of the task definition to run in your service. If a revision is not specified, the latest ACTIVE revision is used. If you modify the task definition with UpdateService, Amazon ECS spawns a task with the new version of the task definition and then stops an old task after the new version is running.
**capacityProviderStrategy**

The capacity provider strategy to update the service to use.

If the service is using the default capacity provider strategy for the cluster, the service can be updated to use one or more capacity providers as opposed to the default capacity provider strategy. However, when a service is using a capacity provider strategy that is not the default capacity provider strategy, the service cannot be updated to use the cluster’s default capacity provider strategy.

A capacity provider strategy consists of one or more capacity providers along with the base and weight to assign to them. A capacity provider must be associated with the cluster to be used in a capacity provider strategy. The PutClusterCapacityProviders API is used to associate a capacity provider with a cluster. Only capacity providers with an `ACTIVE` or `UPDATING` status can be used.

If specifying a capacity provider that uses an Auto Scaling group, the capacity provider must already be created. New capacity providers can be created with the CreateCapacityProvider API operation.

To use a AWS Fargate capacity provider, specify either the `FARGATE` or `FARGATE_SPOT` capacity providers. The AWS Fargate capacity providers are available to all accounts and only need to be associated with a cluster to be used.

The PutClusterCapacityProviders API operation is used to update the list of available capacity providers for a cluster after the cluster is created.

**deploymentConfiguration**

Optional deployment parameters that control how many tasks run during the deployment and the ordering of stopping and starting tasks.

**networkConfiguration**

No description can be found.

**placementConstraints**

An array of task placement constraint objects to update the service to use. If no value is specified, the existing placement constraints for the service will remain unchanged. If this value is specified, it will override any existing placement constraints defined for the service. To remove all existing placement constraints, specify an empty array.

You can specify a maximum of 10 constraints per task (this limit includes constraints in the task definition and those specified at runtime).

**placementStrategy**

The task placement strategy objects to update the service to use. If no value is specified, the existing placement strategy for the service will remain unchanged. If this value is specified, it will override the existing placement strategy defined for the service. To remove an existing placement strategy, specify an empty object.

You can specify a maximum of five strategy rules per service.
platformVersion

The platform version on which your tasks in the service are running. A platform version is only specified for tasks using the Fargate launch type. If a platform version is not specified, the LATEST platform version is used by default. For more information, see AWS Fargate Platform Versions in the Amazon Elastic Container Service Developer Guide.

forceNewDeployment

Whether to force a new deployment of the service. Deployments are not forced by default. You can use this option to trigger a new deployment with no service definition changes. For example, you can update a service’s tasks to use a newer Docker image with the same image/tag combination (my_image:latest) or to roll Fargate tasks onto a newer platform version.

healthCheckGracePeriodSeconds

The period of time, in seconds, that the Amazon ECS service scheduler should ignore unhealthy Elastic Load Balancing target health checks after a task has first started. This is only valid if your service is configured to use a load balancer. If your service’s tasks take a while to start and respond to Elastic Load Balancing health checks, you can specify a health check grace period of up to 2,147,483,647 seconds. During that time, the Amazon ECS service scheduler ignores the Elastic Load Balancing health check status. This grace period can prevent the ECS service scheduler from marking tasks as unhealthy and stopping them before they have time to come up.

description

Modifies which task set in a service is the primary task set. Any parameters that are updated on the primary task set in a service will transition to the service. This is used when a service uses the EXTERNAL deployment controller type. For more information, see Amazon ECS Deployment Types in the Amazon Elastic Container Service Developer Guide.

Usage

```r
ecs_update_service_primary_task_set(
    cluster = NULL,
    service = NULL,
    primaryTaskSet = NULL,
    simplify = TRUE,
    others = list(),
    print_on_error = aws_get_print_on_error(),
    retry_time = aws_get_retry_time(),
    network_timeout = aws_get_network_timeout(),
    region = aws_get_region()
)
```
eca_update_service_primary_task_set

**Arguments**

- **cluster**: Character. The short name or full Amazon Resource Name (ARN) of the cluster that hosts the service that the...
- **service**: Character. The short name or full Amazon Resource Name (ARN) of the service that the task set exists in.
- **primaryTaskSet**: Character. The short name or full Amazon Resource Name (ARN) of the task set to set as the primary task set...
- **simplify**: Logical. Whether to simplify the result and handle nextToken in the response[optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent retry_time times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.

**Value**

A list object or a character vector

- **cluster**: The short name or full Amazon Resource Name (ARN) of the cluster that hosts the service that the task set exists in.

- **service**: The short name or full Amazon Resource Name (ARN) of the service that the task set exists in.

- **primaryTaskSet**: The short name or full Amazon Resource Name (ARN) of the task set to set as the primary task set in the deployment.
**Description**

Modifies a task set. This is used when a service uses the EXTERNAL deployment controller type. For more information, see Amazon ECS Deployment Types in the Amazon Elastic Container Service Developer Guide.

**Usage**

```r
escs_update_task_set(
  cluster = NULL,
  service = NULL,
  taskSet = NULL,
  scale = NULL,
  simplify = TRUE,
  others = list(),
  print_on_error = aws_get_print_on_error(),
  retry_time = aws_get_retry_time(),
  network_timeout = aws_get_network_timeout(),
  region = aws_get_region()
)
```

**Arguments**

- **cluster**: Character. The short name or full Amazon Resource Name (ARN) of the cluster that hosts the service that the...
- **service**: Character. The short name or full Amazon Resource Name (ARN) of the service that the task set exists in.
- **taskSet**: Character. The short name or full Amazon Resource Name (ARN) of the task set to update.
- **scale**: No description can be found.
- **simplify**: Logical. Whether to simplify the result and handle `nextToken` in the response[optional]
- **others**: Named list. The parameters that are not included in the function parameters and need to be added into the request[optional]
- **print_on_error**: Logical. Whether to show an error message when a network error occurs.
- **retry_time**: Integer. Number of retries for a REST request when encounter the network issue. If the request has been sent `retry_time` times but still not be able to get the response, an error will be thrown.
- **network_timeout**: Numeric. Number of seconds to wait for a REST response until giving up. Can not be less than 1 ms.
- **region**: Character. The region of the AWS service.
list_to_array

Value
A list object or a character vector

cluster
The short name or full Amazon Resource Name (ARN) of the cluster that hosts the service that the task set exists in.

service
The short name or full Amazon Resource Name (ARN) of the service that the task set exists in.

taskSet
The short name or full Amazon Resource Name (ARN) of the task set to update.

scale
No description can be found.

list_to_array Utility functions

Description
Utility functions. list_to_array converts a list or a vector to an Array object in AWS Documentation. list_to_filter converts a filter list to an Array object.

Usage
list_to_array(prefix, x)

list_to_filter(x)

Arguments
prefix Character, the name of the parameter
x List or Vector, the value of the parameter

Value
A list object

Examples
vpc_id = c("vpc-081ec835f3", "vpc-0ee975135d", "vpc-06e4ab6c6c")
list_to_array("VpcId", vpc_id)

filter = list("dhcp-options-id" = c("dopt-7a8b9c2d", "dopt-2b2a3d3c"), state="available")
list_to_filter(filter)
Index

aws_get_access_key_id
  (aws_set_credentials), 12
aws_get_credentials
  (aws_set_credentials), 12
aws_get_network_timeout
  (aws_set_retry_time), 14
aws_get_print_on_error
  (aws_set_retry_time), 14
aws_get_region (aws_set_credentials), 12
aws_get_retry_time
  (aws_set_retry_time), 14
aws_get_secret_access_key
  (aws_set_credentials), 12
aws_list_regions (aws_set_credentials), 12
aws_set_access_key_id
  (aws_set_credentials), 12
aws_set_credentials, 12
aws_set_network_timeout
  (aws_set_retry_time), 14
aws_set_print_on_error
  (aws_set_retry_time), 14
aws_set_region (aws_set_credentials), 12
aws_set_retry_time, 14
aws_set_secret_access_key
  (aws_set_credentials), 12
CommonDoc, 15
ec2_accept_reserved_instances_exchange_quote,
  15
ec2_accept_transit_gateway_multicast_domain_associations,
  17
ec2_accept_transit_gateway_peering_attachment
  18
ec2_accept_transit_gateway_vpc_attachment,
  20
ec2_accept_vpc_endpoint_connections,
  21
ec2_accept_vpc_peering_connection, 22
ec2_advertise_byoip_cidr, 23
ec2_allocate_address, 24
ec2_allocate_hosts, 27
ec2_apply_security_groups_to_client_vpn_target_network,
  29
ec2_assign_ipv6_addresses, 31
ec2_assign_private_ip_addresses, 32
ec2_associate_address, 34
ec2_associate_client_vpn_target_network,
  36
ec2_associate_dhcp_options, 37
ec2_associate_enclave_certificate_iam_role,
  38
ec2_associate_iam_instance_profile, 40
ec2_associate_route_table, 41
ec2_associate_subnet_cidr_block, 42
ec2_associate_transit_gateway_multicast_domain,
  43
ec2_associate_transit_gateway_route_table,
  45
ec2_associate_vpc_cidr_block, 46
ec2_attach_classic_link_vpc, 48
ec2_attach_internet_gateway, 50
ec2_attach_network_interface, 51
ec2_attach_volume, 53
ec2_attach_vpn_gateway, 54
ec2_authorize_client_vpn_ingress, 55
ec2_authorize_security_group_egress,
  57
ec2_authorize_security_group_ingress,
  58
ec2_bundle_instance, 62
c2_cancel_bundle_task, 64
c2_cancel_capacity_reservation, 65
c2_cancel_conversion_task, 66
c2_cancel_export_task, 67
c2_cancel_import_task, 68
c2_cancel_reserved_instances_listing,
<table>
<thead>
<tr>
<th>Command</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ec2_cancel_spot_fleet_requests</td>
<td>70</td>
</tr>
<tr>
<td>ec2_cancel_spot_instance_requests</td>
<td>72</td>
</tr>
<tr>
<td>ec2_confirm_product_instance</td>
<td>73</td>
</tr>
<tr>
<td>ec2_copy_fpga_image</td>
<td>74</td>
</tr>
<tr>
<td>ec2_copy_image</td>
<td>76</td>
</tr>
<tr>
<td>ec2_copy_snapshot</td>
<td>79</td>
</tr>
<tr>
<td>ec2_create_capacity_reservation</td>
<td>82</td>
</tr>
<tr>
<td>ec2_create_carrier_gateway</td>
<td>85</td>
</tr>
<tr>
<td>ec2_create_client_vpn_endpoint</td>
<td>87</td>
</tr>
<tr>
<td>ec2_create_client_vpn_route</td>
<td>91</td>
</tr>
<tr>
<td>ec2_create_customer_gateway</td>
<td>93</td>
</tr>
<tr>
<td>ec2_create_default_subnet</td>
<td>95</td>
</tr>
<tr>
<td>ec2_create_default_vpc</td>
<td>96</td>
</tr>
<tr>
<td>ec2_create_dhcp_options</td>
<td>97</td>
</tr>
<tr>
<td>ec2_create_egress_only_internet_gateway</td>
<td>98</td>
</tr>
<tr>
<td>ec2_create_fleet</td>
<td>100</td>
</tr>
<tr>
<td>ec2_create_flow_logs</td>
<td>103</td>
</tr>
<tr>
<td>ec2_create_fpga_image</td>
<td>106</td>
</tr>
<tr>
<td>ec2_create_image</td>
<td>108</td>
</tr>
<tr>
<td>ec2_create_instance_export_task</td>
<td>110</td>
</tr>
<tr>
<td>ec2_create_internet_gateway</td>
<td>112</td>
</tr>
<tr>
<td>ec2_create_key_pair</td>
<td>113</td>
</tr>
<tr>
<td>ec2_create_launch_template</td>
<td>114</td>
</tr>
<tr>
<td>ec2_create_launch_template_version</td>
<td>116</td>
</tr>
<tr>
<td>ec2_create_local_gateway_route</td>
<td>118</td>
</tr>
<tr>
<td>ec2_create_local_gateway_route_table_vpc_association</td>
<td>119</td>
</tr>
<tr>
<td>ec2_create_managed_prefix_list</td>
<td>121</td>
</tr>
<tr>
<td>ec2_create_nat_gateway</td>
<td>123</td>
</tr>
<tr>
<td>ec2_create_network_acl</td>
<td>125</td>
</tr>
<tr>
<td>ec2_create_network_acl_entry</td>
<td>126</td>
</tr>
<tr>
<td>ec2_create_network_insights_path</td>
<td>128</td>
</tr>
<tr>
<td>ec2_create_network_interface</td>
<td>131</td>
</tr>
<tr>
<td>ec2_create_network_interface_permission</td>
<td>133</td>
</tr>
<tr>
<td>ec2_create_placement_group</td>
<td>135</td>
</tr>
<tr>
<td>ec2_create_reserved_instances_listing</td>
<td>137</td>
</tr>
<tr>
<td>ec2_create_route</td>
<td>138</td>
</tr>
<tr>
<td>ec2_create_route_table</td>
<td>141</td>
</tr>
<tr>
<td>ec2_create_security_group</td>
<td>142</td>
</tr>
<tr>
<td>ec2_create_snapshot</td>
<td>144</td>
</tr>
<tr>
<td>ec2_create_snapshots</td>
<td>146</td>
</tr>
<tr>
<td>ec2_create_spot_datafeed_subscription</td>
<td>148</td>
</tr>
<tr>
<td>ec2_create_subnet</td>
<td>149</td>
</tr>
<tr>
<td>ec2_create_tags</td>
<td>151</td>
</tr>
<tr>
<td>ec2_create_traffic_mirror_filter</td>
<td>153</td>
</tr>
<tr>
<td>ec2_create_traffic_mirror_filter_rule</td>
<td>154</td>
</tr>
<tr>
<td>ec2_create_traffic_mirror_session</td>
<td>157</td>
</tr>
<tr>
<td>ec2_create_traffic_mirror_target</td>
<td>159</td>
</tr>
<tr>
<td>ec2_create_transit_gateway</td>
<td>161</td>
</tr>
<tr>
<td>ec2_create_transit_gateway_connect</td>
<td>163</td>
</tr>
<tr>
<td>ec2_create_transit_gateway_connect_peer</td>
<td>164</td>
</tr>
<tr>
<td>ec2_create_transit_gateway_multicast_domain</td>
<td>166</td>
</tr>
<tr>
<td>ec2_create_transit_gateway_peering_attachment</td>
<td>168</td>
</tr>
<tr>
<td>ec2_create_transit_gateway_prefix_list_reference</td>
<td>169</td>
</tr>
<tr>
<td>ec2_create_transit_gateway_route</td>
<td>171</td>
</tr>
<tr>
<td>ec2_create_transit_gateway_route_table</td>
<td>173</td>
</tr>
<tr>
<td>ec2_create_transit_gateway_vpc_attachment</td>
<td>174</td>
</tr>
<tr>
<td>ec2_create_volume</td>
<td>176</td>
</tr>
<tr>
<td>ec2_create_vpc</td>
<td>179</td>
</tr>
<tr>
<td>ec2_create_vpc_endpoint</td>
<td>182</td>
</tr>
<tr>
<td>ec2_create_vpc_endpoint_connection_notification</td>
<td>185</td>
</tr>
<tr>
<td>ec2_create_vpc_peering_connection</td>
<td>188</td>
</tr>
<tr>
<td>ec2_create_vpn_connection</td>
<td>190</td>
</tr>
<tr>
<td>ec2_create_vpn_connection_route</td>
<td>192</td>
</tr>
<tr>
<td>ec2_create_vpn_gateway</td>
<td>193</td>
</tr>
<tr>
<td>ec2_delete_carrier_gateway</td>
<td>195</td>
</tr>
<tr>
<td>ec2_delete_client_vpn_endpoint</td>
<td>196</td>
</tr>
<tr>
<td>ec2_delete_client_vpn_route</td>
<td>197</td>
</tr>
<tr>
<td>ec2_delete_customer_gateway</td>
<td>199</td>
</tr>
<tr>
<td>ec2_delete_dhcp_options</td>
<td>200</td>
</tr>
<tr>
<td>ec2_delete_egress_only_internet_gateway</td>
<td>201</td>
</tr>
<tr>
<td>ec2_delete_fleets</td>
<td>202</td>
</tr>
<tr>
<td>ec2_delete_flow_logs</td>
<td>204</td>
</tr>
<tr>
<td>ec2_delete_fpga_image</td>
<td>205</td>
</tr>
<tr>
<td>ec2_delete_internet_gateway</td>
<td>206</td>
</tr>
<tr>
<td>ec2_delete_key_pair</td>
<td>207</td>
</tr>
<tr>
<td>ec2_delete_launch_template</td>
<td>208</td>
</tr>
<tr>
<td>ec2_delete_launch_template_versions</td>
<td>210</td>
</tr>
</tbody>
</table>
ec2_delete_local_gateway_route, 211
ec2_delete_local_gateway_route_table_vpc_association, 213
ec2_delete_managed_prefix_list, 214
ec2_delete_nat_gateway, 215
ec2_delete_network_acl, 216
ec2_delete_network_acl_entry, 217
ec2_delete_network_insights_analysis, 219
ec2_delete_network_insights_path, 220
ec2_delete_network_interface, 221
ec2_delete_network_interface_permission, 222
ec2_delete_placement_group, 224
ec2_delete_queued_reserved_instances, 225
ec2_delete_route, 226
ec2_delete_route_table, 228
ec2_delete_security_group, 229
ec2_delete_snapshot, 230
ec2_delete_spot_datafeed_subscription, 231
ec2_delete_subnet, 232
ec2_delete_tags, 234
ec2_delete_traffic_mirror_filter, 235
ec2_delete_traffic_mirror_filter_rule, 236
ec2_delete_traffic_mirror_session, 237
ec2_delete_traffic_mirror_target, 239
ec2_delete_transit_gateway, 240
ec2_delete_transit_gateway_connect, 241
ec2_delete_transit_gateway_connect_peer, 242
ec2_delete_transit_gateway_multicast_domain, 243
ec2_delete_transit_gateway_peering_attachment, 245
ec2_delete_transit_gateway_prefix_list_reference, 246
ec2_delete_transit_gateway_route, 247
ec2_delete_transit_gateway_route_table, 248
ec2_delete_transit_gateway_vpc_attachment, 250
ec2_delete_volume, 251
ec2_delete_vpc, 252
ec2_delete_vpc_endpoint_service_configurations, 254
ec2_delete_vpc_endpoints, 253
ec2_delete_vpc_peering_connection, 257
ec2_delete_vpn_connection, 258
ec2_delete_vpn_connection_route, 259
ec2_delete_vpn_gateway, 260
ec2_deprovision_byoip_cidr, 261
ec2_deregister_image, 263
ec2_deregister_instance_event_notification_attributes, 264
ec2_deregister_transit_gateway_multicast_group_members, 265
ec2_deregister_transit_gateway_multicast_group_sources, 266
ec2_describe_account_attributes, 268
ec2_describe_addresses, 269
ec2_describe_addresses_attribute, 271
ec2_describe_aggregate_id_format, 272
ec2_describe_availability_zones, 273
ec2_describe_bundle_tasks, 276
ec2_describe_byoip_cidrs, 277
ec2_describe_capacity_reservations, 278
ec2_describeCarrier_gateways, 281
ec2_describeClassic_link_instances, 283
ec2_describe_client_vpn_authorization_rules, 285
ec2_describe_client_vpn_connections, 286
ec2_describe_client_vpn_endpoints, 288
ec2_describe_client_vpn_routes, 289
ec2_describe_client_vpn_target_networks, 291
e2_describe_coip_pools, 293
e2_describe_conversion_tasks, 294
e2_describe_customer_gateways, 296
e2_describe_dhcp_options, 297
e2_describe_egress_only_internet_gateways, 299
e2_describe_elastic_gpus, 301
e2_describe_export_image_tasks, 302
e2_describe_export_tasks, 304
e2_describe_fast_snapshot_restores, 305
e2_describe_fleet_history, 308
ec2_describe_fleet_instances, 310
ec2_describe_fleets, 306
ec2_describe_flow_logs, 311
ec2_describe_fpga_image_attribute, 315
ec2_describe_fpga_images, 313
ec2_describe_host_reservation_offerings, 320
ec2_describe_host_reservations, 318
ec2_describe_hosts, 316
ec2_describe_iam_instance_profile_associations, 322
ec2_describe_id_format, 324
ec2_describe_identity_id_format, 323
ec2_describe_image_attribute, 328
ec2_describe_images, 325
ec2_describe_import_image_tasks, 329
ec2_describe_import_snapshot_tasks, 331
ec2_describe_instance_attribute, 337
ec2_describe_instance_credit_specifications, 338
ec2_describe_instance_event_notification_attributes, 340
ec2_describe_instance_status, 341
ec2_describe_instance_type_offering, 347
ec2_describe_instance_types, 343
ec2_describe_instances, 332
ec2_describe_internet_gateways, 348
ec2_describe_ipv6_pools, 350
ec2_describe_key_pairs, 352
ec2_describe_launch_template_versions, 355
ec2_describe_launch_templates, 353
ec2_describe_local_gateway_route_table_virtual_interface_group_associations, 361
ec2_describe_local_gateway_route_table_virtual_interface_groups, 363
ec2_describe_local_gateway_route_table_vpc_associations, 360
ec2_describe_local_gateway_routes, 366
ec2_describe_local_gateway_virtual_interfaces, 365
ec2_describe_local_gateways, 358
ec2_describe_managed_prefix_lists, 368
ec2_describe_moving_addresses, 370
ec2_describe_nat_gateways, 371
ec2_describe_network_acls, 373
ec2_describe_network_insights_analyses, 375
ec2_describe_network_insights_paths, 377
ec2_describe_network_interface_attribute, 382
ec2_describe_network_interface_permissions, 383
ec2_describe_network_interfaces, 379
ec2_describe_placement_groups, 385
ec2_describe_prefix_lists, 386
ec2_describe_principal_id_format, 388
ec2_describe_public_ipv4_pools, 389
ec2_describe_regions, 391
ec2_describe_reserved_instances, 392
ec2_describe_reserved_instances_listings, 395
ec2_describe_reserved_instances_modifications, 396
ec2_describe_reserved_instances_offerings, 398
ec2_describe_route_tables, 401
ec2_describe_scheduled_instance_availability, 406
ec2_describe_scheduled_instances, 404
ec2_describe_security_group_references, 411
ec2_describe_security_groups, 408
ec2_describe_snapshot_attribute, 414
ec2_describe_snapshots, 412
ec2_describe_spot_datafeed_subscription, 416
ec2_describe_spot_fleet_instances, 417
ec2_describe_spot_fleet_request_history, 420
ec2_describe_spot_fleet_requests, 418
ec2_describe_spot_instance_requests, 421
ec2_describe_spot_price_history, 424
ec2_describe_stale_security_groups, 427
ec2_describe_subnets, 428
ec2_describe_tags, 430
ec2_describe_traffic_mirror_filters, 432
ec2_describe_traffic_mirror_sessions, 434
INDEX

ec2_describe_traffic_mirror_targets, 435
ec2_describe_transit_gateway_attachments, 439
ec2_describe_transit_gateway_connect_peers, 443
ec2_describe_transit_gateway_connects, 441
ec2_describe_transit_gateway_multicast_domains, 444
ec2_describe_transit_gateway_peering_attachments, 446
ec2_describe_transit_gateway_route_tables, 448
ec2_describe_transit_gateway_vpc_attachments, 449
ec2_describe_transit_gateways, 437
ec2_describe_volume_attribute, 455
ec2_describe_volume_status, 457
ec2_describe_volumes, 451
ec2_describe_volumes_modifications, 453
ec2_describe_vpc_attribute, 461
ec2_describe_vpc_classic_link, 462
ec2_describe_vpc_classic_link_dns_support, 464
ec2_describe_vpc_endpoint_connection_notifications, 468
ec2_describe_vpc_endpoint_connections, 467
ec2_describe_vpc_endpoint_service_configurations, 472
ec2_describe_vpc_endpoint_service_permissions, 473
ec2_describe_vpc_endpoint_services, 470
ec2_describe_vpcs, 459
ec2_describe_vpn_connections, 477
ec2_describe_vpn_gateways, 479
ec2_detach_classic_link_vpc, 481
ec2_detach_internet_gateway, 482
ec2_detach_network_interface, 483
ec2_detach_volume, 485
ec2_detach_vpn_gateway, 486
ec2_disable_ebs_encryption_by_default, 488
ec2_disable_fast_snapshot_restores, 489
ec2_disable_transit_gateway_route_tablePropagation, 490
ec2_disable_vgw_route_propagation, 491
ec2_disable_vpc_classic_link, 493
ec2_disable_vpc_classic_link_dns_support, 494
ec2_disassociate_address, 495
ec2_disassociate_client_vpn_targetNetwork, 496
ec2_disassociate_enclave_certificate_iam_role, 497
ec2_disassociate_route_table, 500
ec2_disassociate_subnet_cidr_block, 501
ec2_disassociate_transit_gateway_multicast_domain, 502
ec2_disassociate_transit_gateway_route_table, 503
ec2_disassociate_vpc_cidr_block, 505
ec2_disable_ebs_encryption_by_default, 506
ec2_enable_ebs_encryption_by_default, 507
ec2_enable_fast_snapshot_restores, 508
ec2_enable_transit_gateway_route_tablePropagation, 509
ec2_enable_vgw_route_propagation, 511
ec2_enable_volume_io, 512
ec2_enable_vpc_classic_link, 513
ec2_enable_vpc_classic_link_dns_support, 514
ec2_export_client_vpn_client_certificate_revocation_list, 515
ec2_export_client_vpn_client_configuration, 516
ec2_export_image, 518
ec2_export_transit_gateway_routes, 519
ec2_get_associated_enclave_certificate_iam_roles, 520
ec2_get_associated_ipv6_pool_cidrs, 521
ec2_get_capacity_reservation_usage, 523
ec2_get_coip_pool_usage, 524
ec2_get_console_output, 526
ec2_get_console_screenshot
ec2_get_default_credit_specification
ec2_get_ebs_default_kms_key_id
ec2_get_ebs_encryption_by_default
ec2_get_groups_for_capacity_reservation
ec2_get_host_reservation_purchase_preview
ec2_get_launch_template_data
ec2_get_managed_prefix_list_associations
ec2_get_managed_prefix_list_entries
ec2_get_password_data
ec2_get_reservedInstances_exchange_quote
ec2_get_transit_gateway_attachment_propagations
ec2_get_transit_gateway_multicast_domain_associations
ec2_get_transit_gateway_prefix_list_references
ec2_get_transit_gateway_route_table_associations
ec2_get_transit_gateway_route_table_propagations
ec2_import_client_vpn_client_certificate_revocation_list
ec2_import_image
ec2_import_instance
ec2_import_key_pair
ec2_import_snapshot
ec2_import_volume
ec2_modify_address_attribute
ec2_modify_availability_zone_group
ec2_modify_capacity_reservation
ec2_modify_client_vpn_endpoint
ec2_modify_default_credit_specification
ec2_modify_ebs_default_kms_key_id
ec2_modify_fleet
ec2_modify_fpga_image_attribute
ec2_modify_hosts
ec2_modify_id_format
ec2_modify_identity_id_format
ec2_modify_image_attribute
ec2_modify_instance_attribute
ec2_modify_instance_capacity_reservation_attributes
ec2_modify_instance_credit_specification
ec2_modify_instance_event_start_time
ec2_modify_instance_metadata_options
ec2_modify_instance_placement
ec2_modify_launch_template
ec2_modify_managed_prefix_list
ec2_modify_network_interface_attribute
ec2_modify_reserved_instances
ec2_modify_snapshot_attribute
ec2_modify_spot_fleet_request
ec2_modify_subnet_attribute
ec2_modify_traffic_mirror_filter_network_services
ec2_modify_traffic_mirror_filter_rule
ec2_modify_traffic_mirror_session
ec2_modify_traffic_mirror_vrf
ec2_modify_transit_gateway
ec2_modify_transit_gateway_prefix_list_reference
ec2_modify_transit_gateway_vpc_attachment
ec2_modify_volume
ec2_modify_vpc_attribute
ec2_modify_vpc_endpoint
ec2_modify_vpc_endpoint_connection_notification
ec2_modify_vpc_endpoint_service_configuration
ec2_modify_vpc_endpoint_service_permissions
ec2_modify_vpc_peering_connection_options
ec2_modify_vpc_tenancy
ec2_modify_vpn_connection
ec2_modify_vpn_connection_options
ec2_modify_vpn_tunnel_certificate
ec2_modify_vpn_tunnel_options
ec2_monitor_instances
ec2_move_address_to_vpc
ec2_provision_byoip_cidr
INDEX

ec2_purchase_host_reservation, 643
ec2_purchase_reserved_instances_offering, 645
ec2_purchase_scheduled_instances, 647
ec2_reboot_instances, 648
ec2_register_image, 649
ec2_register_instance_event_notification_attributes, 652
ec2_register_transit_gateway_multicast_group, 653
ec2_register_transit_gateway_multicast_group_sources, 655
ec2_reject_transit_gateway_multicast_domain_associations, 656
ec2_reject_transit_gateway_peering_attachment, 658
ec2_reject_transit_gateway_vpc_attachment, 659
ec2_reject_vpc_endpoint_connections, 660
ec2_reject_vpc_peering_connection, 661
ec2_release_address, 663
ec2_release_hosts, 664
ec2_replace_iam_instance_profile_association, 665
ec2_replace_network_acl_association, 666
ec2_replace_network_acl_entry, 668
ec2_replace_route, 670
ec2_replace_route_table_association, 673
ec2_replace_transit_gateway_route, 674
ec2_report_instance_status, 676
ec2_request Spot Fleet, 678
ec2_request_spot_instances, 679
ec2_reset_address_attribute, 683
ec2_reset_ebs_default_kms_key_id, 684
ec2_reset_fpga_image_attribute, 685
ec2_reset_image_attribute, 687
ec2_reset_instance_attribute, 688
ec2_reset_network_interface_attribute, 689
ec2_reset_snapshot_attribute, 691
ec2_restore_address_to_classic, 692
ec2_restore_managed_prefix_list_version, 693
ec2_revoke_client_vpn_ingress, 695
ec2_revoke_security_group_egress, 696
ec2_revoke_security_group_ingress, 699
ec2_run_instances, 701
ec2_run_scheduled_instances, 709
ec2_search_local_gateway_routes, 711
ec2_search_transit_gateway_multicast_groups, 712
ec2_send_diagnostic_interrupt, 716
ec2_start_instances, 717
ec2_start_network_insights_analysis, 718
ec2_start_vpc_endpoint_service_private_dns_verification, 720
ec2_stop_instances, 721
ec2_terminate_client_vpn_connections, 722
ec2_terminate_instances, 724
ec2_unassign_ipv6_addresses, 725
ec2_unassign_private_ip_addresses, 726
ec2_unmonitor_instances, 727
ec2_update_security_group_rule_descriptions_egress, 729
ec2_update_security_group_rule_descriptions_ingress, 730
e2c_withdraw_byoip_cidr, 732
ecs_create_capacity_provider, 733
ecs_create_cluster, 734
ecs_create_service, 737
ecs_create_task_set, 743
ecs_delete_account_setting, 747
ecs_delete_attributes, 748
ecs_delete_capacity_provider, 749
ecs_delete_cluster, 750
ecs_delete_service, 751
ecs_delete_task_set, 753
ecs_deregister_container_instance, 754
ecs_deregister_task_definition, 756
ecs_describe_capacity_providers, 757
ecs_describe_clusters, 758
ecs_describe_container_instances, 760
ecs_describe_services, 761
ecs_describe_task_definitions, 764
ecs_describe_task_sets, 765
ecs_describe_tasks, 762
ecs_discover_poll_endpoint, 766
ecs_list_account_settings, 767
ecs_list_attributes, 769
ecs_list_clusters, 771
ecs_list_container_instances, 772
ecs_list_services, 773
ecs_list_tags_for_resource, 775
ecs_list_task_definition_families, 779
ecs_list_task_definitions, 778
ecs_list_tasks, 776
ecs_put_account_setting, 781
ecs_put_account_setting_default, 782
ecs_put_attributes, 783
ecs_put_cluster_capacity_providers, 785
ecs_register_container_instance, 786
ecs_register_task_definition, 789
ecs_run_task, 794
ecs_start_task, 799
ecs_stop_task, 802
ecs_submit_attachment_state_changes, 803
ecs_submit_container_state_change, 804
ecs_submit_task_state_change, 806
ecs_tag_resource, 808
ecs_untag_resource, 810
ecs_update_capacity_provider, 811
ecs_update_cluster_settings, 812
ecs_update_container_agent, 813
ecs_update_container_instances_state, 814
ecs_update_service, 816
ecs_update_service_primary_task_set, 819
ecs_update_task_set, 821
list_to_array, 822
list_to_filter (list_to_array), 822