Package ‘aws.sqs’

July 4, 2017

Type Package
Title AWS SQS Client Package
Version 0.1.10
Date 2017-07-04
Description A simple client package for the Amazon Web Services ('AWS') Simple Queue Service ('SQS') <https://aws.amazon.com/sqs/> 'API'.
License GPL (>= 2)
Imports httr, xml2, jsonlite, aws.signature (>= 0.3.4)
Suggests testthat
URL https://github.com/cloudyr/aws.sqs
BugReports https://github.com/cloudyr/aws.sqs/issues
RoxygenNote 6.0.1
NeedsCompilation no
Author Thomas J. Leeper [aut, cre]
Maintainer Thomas J. Leeper <thosjleeper@gmail.com>
Repository CRAN
Date/Publication 2017-07-04 20:31:43 UTC

R topics documented:

aws.sqs-package ......................................................... 2
add_queue_permission .................................................. 2
create_queue ............................................................... 4
get_queue_attrs ............................................................ 5
get_queue_url ............................................................... 6
list_queues ................................................................. 7
purge_queue ................................................................. 8
receive_msg ............................................................... 9
send_msg ................................................................. 10
sqsHTTP ................................................................. 12
visibility ................................................................. 13
add_queue_permission

Description

A simple client package for the Amazon Web Services (AWS) Simple Queue Service (SQS) REST API.

Author(s)

Thomas J. Leeper <thosjleeper@gmail.com>

Examples

```r
## not run:
# list current queues
list_queues()

# create a queue
queue <- create_queue("ExampleQueue")
get_queue_url("ExampleQueue")

# send message to queue
send_msg("ExampleQueue", "This is a test message")
# receive a message
(m <- receive_msg("ExampleQueue", timeout = 0))

# delete a message from queue
delete_msg("ExampleQueue", m$ReceiptHandle[1])

# delete queue
delete_queue("ExampleQueue")

## end(not run)
```

add_queue_permission  Change queue permissions

Description

Add or remove queue permissions
add_queue_permission

Usage

add_queue_permission(queue, label, principal, action, query = NULL, ...)

remove_queue_permission(queue, label, query = NULL, ...)

Arguments

queue A character string containing a queue URL, or the name of the queue.
label A character string containing a unique label for the permission.
principal A character vector containing the AWS account number of the principal who
will be given permission. Principals and actions must be specified one-to-one or
one-to-many.
action A character vector containing an SQS permission. Valid values include: "*",
query A list specifying additional query arguments to be passed to the query argument
of sqsHTTP.
... Additional arguments passed to sqsHTTP.

Details

Add or removes a permission from an SQS queue.

Value

A list.

Author(s)

Thomas J. Leeper

References

AddPermission RemovePermission

See Also

link{create_queue} get_queue_attrs set_queue_attrs
create_queue  

Create or delete a queue

Description
Create or delete an SQS queue

Usage
create_queue(name, attributes = NULL, query = NULL, ...)
delete_queue(queue, query = NULL, ...)

Arguments
name   A character string containing a name for the queue.
attributes      Currently ignored
query    A list specifying additional query arguments to be passed to the query argument of sqsHTTP.
...        Additional arguments passed to sqsHTTP.
queue    A character string containing a queue URL, or the name of the queue.

Details
create_queue creates a new SQS queue; delete_queue deletes a queue.

Value
If successful, create_queue returns a character string containing an SQS Queue URL and delete_queue returns a logical TRUE. Otherwise, a data structure of class "aws_error" containing any error message(s) from AWS and information about the request attempt.

Author(s)
Thomas J. Leeper

References
CreateQueue DeleteQueue

See Also
get_queueAttrs set_queueAttrs link(purge_queue)
get_queueAttrs

Examples

```r
# Not run:
# list current queues
list_queues()

# create a queue
create_queue("ExampleQueue")
get_queue_url("ExampleQueue")

# send message to queue
send_msg("ExampleQueue", "This is a test message")

# receive a message
receive_msg("ExampleQueue", timeout = 0)

# delete a message from queue
delete_msg("ExampleQueue", m$ReceiptHandle[1])

# delete queue
delete_queue("ExampleQueue")

# End(Not run)
```

## get_queueAttrs

### Queue Attributes

#### Description

Get and set queue attributes

#### Usage

```r
get_queueAttrs(queue, attributes = "All", query = NULL, ...)

set_queueAttrs(queue, attributes, query = NULL, ...)
```

#### Arguments

- `queue`: A character string containing a queue URL, or the name of the queue.
- `query`: A list specifying additional query arguments to be passed to the query argument of `sqsHTTP`.
- `...`: Additional arguments passed to `sqsHTTP`. 
get_queue_url

Details

Get or set the attributes for a queue.

Value

For get_queue_attr, a data.frame list of queue attributes. Otherwise, a data structure of class
"aws_error" containing any error message(s) from AWS and information about the request attempt.

For set_queue_attr, a logical TRUE if operation was successful. Otherwise, a data structure of
class "aws_error" containing any error message(s) from AWS and information about the request
attempt.

Author(s)

Thomas J. Leeper

References

ReceiveMessage

See Also

link{create_queue}

get_queue_url

Get a queue URL

Description

Retrieves the URL for an SQS queue by its name.

Usage

get_queue_url(name, owner = NULL, query = NULL, ...)

Arguments

name A character string containing the name of the queue.

owner A character string containing the AWS Account ID that created the queue.

query A list specifying additional query arguments to be passed to the query argument
of sqsHTTP.

... Additional arguments passed to sqsHTTP.

Value

If successful, a character string containing an SQS Queue URL. Otherwise, a data structure of class
"aws_error" containing any error message(s) from AWS and information about the request attempt.
list_queues

Author(s)

Thomas J. Leeper

References

GetQueueURL

See Also

link{create_queue} link{delete_queue} get_queueAttrs set_queueAttrs

Description

List SQS queues

Usage

list_queues(starts_with = NULL, query = NULL, ...)

Arguments

starts_with An optional character string describing the beginning of the name of queues to retrieve.
query A list specifying additional query arguments to be passed to the query argument of sqsHTTP.
... Additional arguments passed to sqsHTTP.

Details

For list_queues, a list of all SQS queue associated with an AWS account, or only those starting with a particular character string.

For deadletter_queues, a list of all SQS queues with a RedrivePolicy queue attribute configured with a dead letter queue.

Value

If successful, a character vector containing queue URLs. Otherwise, a data structure of class “aws_error” containing any error message(s) from AWS and information about the request attempt.

Author(s)

Thomas J. Leeper
purge_queue

References

ListQueues ListDeadLetterSourceQueues

Examples

```r
## not run:
# list current queues
list_queues()

## End(Not run)
```

---

**purge_queue**

**Purge a queue**

### Description

Purge an SQS queue of its message

### Usage

```r
purge_queue(queue, query = NULL, ...)
```

### Arguments

- `queue` A character string containing a queue URL, or the name of the queue.
- `query` A list specifying additional query arguments to be passed to the `query` argument of `sqsHTTP`.
- `...` Additional arguments passed to `sqsHTTP`.

### Details

Purge an SQS queue of its messages, without deleting it. Use `delete_queue` to delete the queue entirely.

### Value

If successful, a logical `TRUE` value. Otherwise, a data structure of class “aws_error” containing any error message(s) from AWS and information about the request attempt.

### Author(s)

Thomas J. Leeper

### References

PurgeQueue
**receive_msg**

**See Also**

`link{create_queue} delete_queue`

---

**receive_msg**  
*Receive messages*

**Description**

Receive one or more messages from an SQS queue.

**Usage**

```r
receive_msg(queue, attributes = NULL, n = 1, timeout = NULL,
             wait = NULL, query = NULL, ...)
```

```r
consume_msg(queue, receive_args = list(), delete_args = list())
```

```r
delete_msg(queue, handle, query = NULL, ...)
```

**Arguments**

- **queue**  
  A character string containing a queue URL, or the name of the queue.

- **attributes**  
  Currently ignored.

- **n**  
  The number of messages to retrieve (maximum 10).

- **timeout**  
  A number of seconds to make the message invisible to subsequent receive_msg requests. This modifies the queue's default visibility timeout. See `visibility` to modify this value after receiving a message.

- **wait**  
  A number of seconds to wait for messages before responding to the request.

- **query**  
  A list specifying additional query arguments to be passed to the query argument of `sqsHTTP`.

- **...**  
  Additional arguments passed to `sqsHTTP`.

- **receive_args**  
  A named list of arguments, other than queue, to be passed to `receive_msg`.

- **delete_args**  
  A named list of arguments, other than queue and handle, to be passed to `delete_msg`.

- **handle**  
  A message handle, as returned by `receive_msg`.

**Details**

`receive_msg` simply receives message(s). `consume_msg` does the same and then deletes the message(s) from the queue. `delete_message` deletes one or more messages from an SQS queue. If a message is not deleted, it remains visible in the queue and will be returned by subsequent calls to `receive_msg`.

**Value**

A data.frame of messages.
Author(s)

Thomas J. Leeper

References

ReceiveMessage DeleteMessage DeleteMessageBatch

See Also

link{send_msg}

Examples

```r
## Not run:
# list current queues
list_queues()

# create a queue
queue <- create_queue("ExampleQueue")
get_queue_url("ExampleQueue")

# send message to queue
send_msg("ExampleQueue", "This is a test message")
# receive a message
(m <- receive_msg("ExampleQueue", timeout = 0))

# delete a message from queue
delete_msg("ExampleQueue", m$ReceiptHandle[1])

# delete queue
delete_queue("ExampleQueue")

## End(Not run)
```

Description

Send a message to an SQS queue

Usage

`send_msg(queue, msg, attributes = NULL, delay = NULL, query = NULL, ...)`
Arguments

queue A character string containing a queue URL, or the name of the queue.

msg A character vector containing one or more message bodies.

attributes Currently ignored. (If msg is of length one, a specification of message attributes. Ignored otherwise.)

delay A numeric value indicating the number of seconds between 0 and 900 to delay a specific message. If NULL, the default value for the queue applies.

query A list specifying additional query arguments to be passed to the query argument of sqsHTTP.

Additional arguments passed to sqsHTTP.

Value

A list of message information, including the MessageId and an MD5 checksum of the message body.

Author(s)

Thomas J. Leeper

References

SendMessage SendMessageBatch

See Also

link{receive_msg} link{delete_msg}

Examples

## Not run:

# list current queues
list_queues()

# create a queue
queue <- create_queue("ExampleQueue")
get_queue_url("ExampleQueue")

# send message to queue
send_msg("ExampleQueue", "This is a test message")

# receive a message
(m <- receive_msg("ExampleQueue", timeout = 0))

# delete a message from queue
delete_msg("ExampleQueue", m$ReceiptHandle[1])

# delete queue
delete_queue("ExampleQueue")
sqsHTTP

### Description

This is the workhorse function to execute calls to the SQS API.

### Usage

```r
sqsHTTP(url = NULL, query = list(),
region = Sys.getenv("AWS_DEFAULT_REGION", "us-east-1"), key = NULL,
secret = NULL, session_token = NULL, ...)
```

### Arguments

- **url**: A character string containing an SQS API endpoint URL.
- **query**: An optional named list containing query string parameters and their character values.
- **region**: A character string containing an AWS region. If missing, the default “us-east-1” is used.
- **key**: A character string containing an AWS Access Key ID. See `locate_credentials`.
- **secret**: A character string containing an AWS Secret Access Key. See `locate_credentials`.
- **session_token**: Optionally, a character string containing an AWS temporary Session Token. See `locate_credentials`.
- **...**: Additional arguments passed to `GET`.

### Details

This function constructs and signs an SQS API request and returns the results thereof, or relevant debugging information in the case of error.

### Value

If successful, a named list. Otherwise, a data structure of class “aws_error” containing any error message(s) from AWS and information about the request attempt.

### Author(s)

Thomas J. Leeper
visibility  

*Change Visibility Timeout*

**Description**

Change the visibility timeout of received message(s)

**Usage**

```r
visibility(queue, handle, timeout = NULL, query = NULL, ...)
```

**Arguments**

- `queue` A character string containing a queue URL, or the name of the queue.
- `handle` A character vector containing one or more message handles, as returned by `receive_msg`.
- `timeout` An integer value indicating the new value of the visibility timeout, in seconds between 0 and 43200, for the message(s).
- `query` A list specifying additional query arguments to be passed to the `query` argument of `sqsHTTP`.
- `...` Additional arguments passed to `sqsHTTP`.

**Details**

Changes the visibility timeout of one or more SQS messages, by their message handle.

**Value**

If successful, a logical `TRUE`. Otherwise, a data structure of class “aws_error” containing any error message(s) from AWS and information about the request attempt.

**Author(s)**

Thomas J. Leeper

**References**

ChangeMessageVisibility

**See Also**

`receive_msg`, `delete_msg`
Index

*Topic package
  aws.sqs-package, 2

add_queue_permission, 2
aws.sqs (aws.sqs-package), 2
aws.sqs-package, 2

consume_msg (receive_msg), 9
create_queue, 4

deadletter_queues (list_queues), 7
delete_msg (receive_msg), 9
delete_queue, 8, 9
delete_queue (create_queue), 4

GET, 12
get_queue_attrs, 3, 4, 5, 7
get_queue_url, 6

list_queues, 7
locate_credentials, 12

purge_queue, 8
receive_msg, 9, 9, 13
remove_queue_permission
  (add_queue_permission), 2

send_msg, 10
set_queue_attrs, 3, 4, 7
set_queue_attrs (get_queue_attrs), 5
sqsHTTP, 3–9, 11, 12, 13

visibility, 9, 13