aws

REST API Reference

Amazon MQ

Amazon MQ: REST API Reference

Table of Contents

Welcome

Welcome to the Amazon MQ REST API reference. In this guide, you can find descriptions of REST resources, example requests, HTTP methods, schemas, parameters, and the errors that the service returns.

Amazon MQ is a managed message broker service for [Apache ActiveMQ](#) and [RabbitMQ](#) that makes it easy to set up and operate message brokers in the cloud. A message broker allows software applications and components to communicate using various programming languages, operating systems, and formal messaging protocols. Amazon MQ works with your existing applications and services without the need to manage, operate, or maintain your own messaging system.

Topics

- [Example REST Request](#)
- [Example REST Response](#)

Additional Information

- [Amazon MQ Product Page](#)
- [*Amazon MQ Developer Guide*](#)
- [Amazon MQ in the *AWS CLI Command Reference*](#)
- [Regions and Endpoints](#)

This document was last published on July 7, 2021.

Example REST Request

The following is an example of an Amazon MQ REST request (and its headers) which creates a new Amazon MQ for ActiveMQ broker:

```
POST /v1/brokers HTTP/1.1
Content-Type: application/json
X-Amz-Date: 20171123T214525Z
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20171128/us-
east-2/mq/aws4_request, SignedHeaders=content-length;content-type;host;x-amz-date,
 Signature=a12bc34567defg89h0ij1234kl56m789no01p2q345r6s789tu01v2w3x4567890
```

```
Host: mq.us-east-2.amazonaws.com
Cache-Control: no-cache


{
    "brokerName": "MyActiveMQBroker",
    "hostInstanceType": "mq.m5.large",
    "engineType": "ActiveMQ",
    "engineVersion": "5.15.8",
    "logs": {
        "general": true,
        "audit": false
    },
    "deploymentMode": "ACTIVE_STANDBY_MULTI_AZ",
    "publiclyAccessible": true,
    "subnetIds": [
        "subnet-12a3b45c",
        "subnet-67d8e90f"
    ],
    "securityGroups": [
        "sg-a1b234cd",
        "sg-e5f678gh"
    ],
    "autoMinorVersionUpgrade": true,
    "users": [{
        "password": "MyPassword456",
        "groups": [
            "admins"
        ],
        "consoleAccess": true,
        "username": "jane.doe"
    }]
}
```

The following is an example of an Amazon MQ REST request (and its headers) which creates a new Amazon MQ for RabbitMQ broker:

```
POST /v1/brokers HTTP/1.1
Content-Type: application/json
X-Amz-Date: 20171123T214525Z
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20171128/us-
east-2/mq/aws4_request, SignedHeaders=content-length;content-type;host;x-amz-date,
 Signature=a12bc34567defg89h0ij1234kl56m789no01p2q345r6s789tu01v2w3x4567890
Host: mq.us-east-2.amazonaws.com
```

```
Cache-Control: no-cache

{
    "brokerName": "MyRabbitMQBroker",
    "hostInstanceType": "mq.m5.large",
    "engineType": "RabbitMQ",
    "engineVersion": "3.8.6",
    "logs": {
        "general": true
    },
    "deploymentMode": "CLUSTER_MULTI_AZ",
    "publiclyAccessible": true,
    "subnetIds": [
        "subnet-15a3b47c",
        "subnet-69d8n90j"
    ],
    "autoMinorVersionUpgrade": true,
    "users": [{
        "password": "MyPassword456",
        "groups": [
            "admins"
        ],
        "consoleAccess": true,
        "username": "jane.doe"
    }]
}
```

Example REST Response

The following is an example of an Amazon MQ REST response (and its headers) which
acknowledges the creation of the new broker:

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 250
Connection: keep-alive
Date: Wed, 28 Nov 2017 12:00:00 GMT
X-Amzn-RequestId: a12bc345-67de-89f0-g12h-345ij6k7l89m
X-Amzn-Trace-Id: sampled=1;root=1-2345a67b-c8defg901hijk2lf3m4nopqr
X-Cache: Miss from cloudfront
Via: 1.1 ab123456cd789ef012g34567890h1i23.cloudfront.net (CloudFront)
X-Amz-Cf-Id: ABCa1D6b2-EcF3dG7e456_fGIgh7JKLMijNO_PQ8RSTUVWX-Y9k0ZA==
```

```
{
    "brokerId" : "b-1234a5b6-78cd-901e-2fgh-3i45j6k17819",
    "brokerArn": "arn:aws:mq:us-
east-2:123456789012:broker:MyBroker:b-1234a5b6-78cd-901e-2fgh-3i45j6k17819"
}
```

Resources

The Amazon MQ REST API includes the following resources.

Topics

- [Broker](#)
- [Broker Engine Types](#)
- [Broker Instance Options](#)
- [Broker Reboot](#)
- [Brokers](#)
- [Brokers broker-id Promote](#)
- [Configuration](#)
- [Configuration Revision](#)
- [Configuration Revisions](#)
- [Configurations](#)
- [Tag](#)
- [User](#)
- [Users](#)

Broker

A broker is a message broker environment running on Amazon MQ. It is the basic building block of Amazon MQ. For more information about the different components of an Amazon MQ broker, see [How Amazon MQ works](#) in the *Amazon MQ Developer Guide*.

URI

/v1/brokers/*broker-id*

HTTP methods

GET

Operation ID: `DescribeBroker`

Returns information about the specified broker.

Path parameters

Name	Type	Required	Description
broker-id	String	True	The unique ID that Amazon MQ generates for the broker.

Responses

Status code	Response model	Description
200	DescribeBrokerOutput	HTTP Status Code 200: OK.
400	Error	HTTP Status Code 400: Bad request due to incorrect input. Correct your request and then retry it.
403	Error	HTTP Status Code 403: Access forbidden. Correct your credentials and then retry your request.
404	Error	HTTP Status Code 404: Resource not found due to incorrect input. Correct your request and then retry it.
500	Error	HTTP Status Code 500: Unexpected internal server error. Retrying your request might resolve the issue.

PUT

Operation ID: `UpdateBroker`

Adds a pending configuration change to a broker.

Path parameters

Name	Type	Required	Description
broker-id	String	True	The unique ID that Amazon MQ generates for the broker.

Responses

Status code	Response model	Description
200	[UpdateBrokerOutput](#)	HTTP Status Code 200: OK.
400	[Error](#)	HTTP Status Code 400: Bad request due to incorrect input. Correct your request and then retry it.
403	[Error](#)	HTTP Status Code 403: Access forbidden. Correct your credentials and then retry your request.
404	[Error](#)	HTTP Status Code 404: Resource not found due to incorrect input. Correct your request and then retry it.
409	[Error](#)	HTTP Status Code 409: Conflict. This broker name

Status code	Response model	Description
		already exists. Retry your request with another name.
500	Error	HTTP Status Code 500: Unexpected internal server error. Retrying your request might resolve the issue.

DELETE

Operation ID: `DeleteBroker`

Deletes a broker. Note: This API is asynchronous.

Path parameters

Name	Type	Required	Description
broker-id	String	True	The unique ID that Amazon MQ generates for the broker.

Responses

Status code	Response model	Description
200	DeleteBrokerOutput	HTTP Status Code 200: OK.
400	Error	HTTP Status Code 400: Bad request due to incorrect input. Correct your request and then retry it.
403	Error	HTTP Status Code 403: Access forbidden. Correct your

Status code	Response model	Description
		credentials and then retry your request.
404	Error	HTTP Status Code 404: Resource not found due to incorrect input. Correct your request and then retry it.
500	Error	HTTP Status Code 500: Unexpected internal server error. Retrying your request might resolve the issue.

OPTIONS

Path parameters

Name	Type	Required	Description
broker-id	String	True	The unique ID that Amazon MQ generates for the broker.

Responses

Status code	Response model	Description
200	None	Default response for CORS method

Schemas

Request bodies

PUT schema

```
{
  "dataReplicationMode": enum,
  "engineVersion": "string",
  "maintenanceWindowStartTime": {
    "dayOfWeek": enum,
    "timeZone": "string",
    "timeOfDay": "string"
  },
  "configuration": {
    "id": "string",
    "revision": integer
  },
  "authenticationStrategy": enum,
  "securityGroups": [
    "string"
  ],
  "ldapServerMetadata": {
    "roleSearchMatching": "string",
    "serviceAccountPassword": "string",
    "roleBase": "string",
    "hosts": [
      "string"
    ],
    "roleName": "string",
    "userBase": "string",
    "roleSearchSubtree": boolean,
    "serviceAccountUsername": "string",
    "userRoleName": "string",
    "userSearchMatching": "string",
    "userSearchSubtree": boolean
  },
  "logs": {
    "general": boolean,
    "audit": boolean
  },
  "hostInstanceType": "string",
  "autoMinorVersionUpgrade": boolean
}
```

Response bodies

DescribeBrokerOutput schema

```
{
  "pendingEngineVersion": "string",
  "pendingAuthenticationStrategy": enum,
  "pendingSecurityGroups": [
    "string"
  ],
  "configurations": {
    "current": {
      "id": "string",
      "revision": integer
    },
    "pending": {
      "id": "string",
      "revision": integer
    },
    "history": [
      {
        "id": "string",
        "revision": integer
      }
    ]
  },
  "brokerState": enum,
  "pendingDataReplicationMode": enum,
  "engineType": enum,
  "brokerInstances": [
    {
      "endpoints": [
        "string"
      ],
      "consoleURL": "string",
      "ipAddress": "string"
    }
  ],
  "hostInstanceType": "string",
  "dataReplicationMode": enum,
  "publiclyAccessible": boolean,
  "logs": {
    "generalLogGroup": "string",
    "general": boolean,
    "audit": boolean,
    "pending": {
```

```
        "general": boolean,
        "audit": boolean
      },
      "auditLogGroup": "string"
    },
    "ldapServerMetadata": {
      "roleSearchMatching": "string",
      "roleBase": "string",
      "hosts": [
        "string"
      ],
      "roleName": "string",
      "userBase": "string",
      "roleSearchSubtree": boolean,
      "serviceAccountUsername": "string",
      "userRoleName": "string",
      "userSearchMatching": "string",
      "userSearchSubtree": boolean
    },
    "subnetIds": [
      "string"
    ],
    "pendingHostInstanceType": "string",
    "engineVersion": "string",
    "brokerId": "string",
    "brokerArn": "string",
    "deploymentMode": enum,
    "maintenanceWindowStartTime": {
      "dayOfWeek": enum,
      "timeZone": "string",
      "timeOfDay": "string"
    },
    "created": "string",
    "authenticationStrategy": enum,
    "users": [
      {
        "pendingChange": enum,
        "username": "string"
      }
    ],
    "tags": {
    },
    "dataReplicationMetadata": {
      "dataReplicationCounterpart": {
```

```
      "brokerId": "string",
      "region": "string"
    },
    "dataReplicationRole": "string"
  },
  "pendingLdapServerMetadata": {
    "roleSearchMatching": "string",
    "roleBase": "string",
    "hosts": [
      "string"
    ],
    "roleName": "string",
    "userBase": "string",
    "roleSearchSubtree": boolean,
    "serviceAccountUsername": "string",
    "userRoleName": "string",
    "userSearchMatching": "string",
    "userSearchSubtree": boolean
  },
  "encryptionOptions": {
    "useAwsOwnedKey": boolean,
    "kmsKeyId": "string"
  },
  "pendingDataReplicationMetadata": {
    "dataReplicationCounterpart": {
      "brokerId": "string",
      "region": "string"
    },
    "dataReplicationRole": "string"
  },
  "storageType": enum,
  "actionsRequired": [
    {
      "actionRequiredCode": "string",
      "actionRequiredInfo": "string"
    }
  ],
  "securityGroups": [
    "string"
  ],
  "brokerName": "string",
  "autoMinorVersionUpgrade": boolean
}
```

UpdateBrokerOutput schema

```
{
  "engineVersion": "string",
  "brokerId": "string",
  "maintenanceWindowStartTime": {
    "dayOfWeek": enum,
    "timeZone": "string",
    "timeOfDay": "string"
  },
  "configuration": {
    "id": "string",
    "revision": integer
  },
  "authenticationStrategy": enum,
  "pendingDataReplicationMode": enum,
  "hostInstanceType": "string",
  "dataReplicationMode": enum,
  "dataReplicationMetadata": {
    "dataReplicationCounterpart": {
      "brokerId": "string",
      "region": "string"
    },
    "dataReplicationRole": "string"
  },
  "pendingDataReplicationMetadata": {
    "dataReplicationCounterpart": {
      "brokerId": "string",
      "region": "string"
    },
    "dataReplicationRole": "string"
  },
  "securityGroups": [
    "string"
  ],
  "ldapServerMetadata": {
    "roleSearchMatching": "string",
    "roleBase": "string",
    "hosts": [
      "string"
    ],
    "roleName": "string",
    "userBase": "string",
    "roleSearchSubtree": boolean,
```

```
      "serviceAccountUsername": "string",
      "userRoleName": "string",
      "userSearchMatching": "string",
      "userSearchSubtree": boolean
    },
    "logs": {
      "general": boolean,
      "audit": boolean
    },
    "autoMinorVersionUpgrade": boolean
}
```

DeleteBrokerOutput schema

```
{
    "brokerId": "string"
}
```

Error schema

```
{
    "errorAttribute": "string",
    "message": "string"
}
```

Properties

ActionRequired

Action required for a broker.

actionRequiredCode

The code you can use to find instructions on the action required to resolve your broker issue.

  Type: string
  Required: False

actionRequiredInfo

Information about the action required to resolve your broker issue.

Type: string
Required: False

AuthenticationStrategy

Optional. The authentication strategy used to secure the broker. The default is SIMPLE.

```
SIMPLE
LDAP
```

BrokerInstance

Returns information about all brokers.

endpoints

The broker's wire-level protocol endpoints.

Type: Array of type string
Required: False

consoleURL

The brokers web console URL.

Type: string
Required: False

ipAddress

The IP address of the Elastic Network Interface (ENI) attached to the broker. Does not apply to RabbitMQ brokers.

Type: string
Required: False

BrokerState

The broker's status.

```
CREATION_IN_PROGRESS
CREATION_FAILED
DELETION_IN_PROGRESS
RUNNING
REBOOT_IN_PROGRESS
CRITICAL_ACTION_REQUIRED
REPLICA
```

BrokerStorageType

The broker's storage type.

> ⚠️ **Important**
>
>   EFS is not supported for RabbitMQ engine type.

```
EBS
EFS
```

ChangeType

The type of change pending for the ActiveMQ user.

```
CREATE
UPDATE
DELETE
```

ConfigurationId

A list of information about the configuration.

id

Required. The unique ID that Amazon MQ generates for the configuration.

Type: string
Required: True

revision

The revision number of the configuration.

> **Type**: integer
> **Required**: False

Configurations

Broker configuration information

current

The broker's current configuration.

> **Type**: [ConfigurationId](#)
> **Required**: False

pending

The broker's pending configuration.

> **Type**: [ConfigurationId](#)
> **Required**: False

history

The history of configurations applied to the broker.

> **Type**: Array of type [ConfigurationId](#)
> **Required**: False

DataReplicationCounterpart

Specifies a broker in a data replication pair.

brokerId

Required. The unique broker id generated by Amazon MQ.

Type: string

Required: True

region

Required. The region of the broker.

Type: string

Required: True

DataReplicationMetadataOutput

The replication details of the data replication-enabled broker. Only returned if dataReplicationMode or pendingDataReplicationMode is set to CRDR.

dataReplicationCounterpart

Describes the replica/primary broker. Only returned if this broker is currently set as a primary or replica in the broker's dataReplicationRole property.

Type: DataReplicationCounterpart

Required: False

dataReplicationRole

Defines the role of this broker in a data replication pair. When a replica broker is promoted to primary, this role is interchanged.

Type: string

Required: True

DataReplicationMode

Specifies whether a broker is a part of a data replication pair.

NONE

CRDR

DeleteBrokerOutput

Returns information about the deleted broker.

brokerId

The unique ID that Amazon MQ generates for the broker.

> **Type**: string
> **Required**: False

DeploymentMode

The broker's deployment mode.

```
SINGLE_INSTANCE
ACTIVE_STANDBY_MULTI_AZ
CLUSTER_MULTI_AZ
```

DescribeBrokerOutput

Returns information about the specified broker.

pendingEngineVersion

The broker engine version to upgrade to. For more information, see the [ActiveMQ version management](#) and the [RabbitMQ version management](#) sections in the Amazon MQ Developer Guide.

> **Type**: string
> **Required**: False

pendingAuthenticationStrategy

The authentication strategy that will be applied when the broker is rebooted. The default is SIMPLE.

> **Type**: [AuthenticationStrategy](#)
> **Required**: False

pendingSecurityGroups

The list of pending security groups to authorize connections to brokers.

Type: Array of type string
Required: False

configurations

The list of all revisions for the specified configuration.

Type: Configurations
Required: False

brokerState

The broker's status.

Type: BrokerState
Required: False

pendingDataReplicationMode

Describes whether this broker will be a part of a data replication pair after reboot.

Type: DataReplicationMode
Required: False

engineType

The type of broker engine. Currently, Amazon MQ supports ACTIVEMQ and RABBITMQ.

Type: EngineType
Required: True

brokerInstances

A list of information about allocated brokers.

Type: Array of type BrokerInstance

Required: False

hostInstanceType

The broker's instance type.

Type: string

Required: False

dataReplicationMode

Describes whether this broker is a part of a data replication pair.

Type: DataReplicationMode

Required: False

publiclyAccessible

Enables connections from applications outside of the VPC that hosts the broker's subnets.

Type: boolean

Required: True

logs

The list of information about logs currently enabled and pending to be deployed for the specified broker.

Type: LogsSummary

Required: False

ldapServerMetadata

The metadata of the LDAP server used to authenticate and authorize connections to the broker.

Type: LdapServerMetadataOutput

Required: False

subnetIds

The list of groups that define which subnets and IP ranges the broker can use from different Availability Zones.

> **Type**: Array of type string
>
> **Required**: False

pendingHostInstanceType

The broker's host instance type to upgrade to. For a list of supported instance types, see Broker instance types.

> **Type**: string
>
> **Required**: False

engineVersion

The broker engine version. For more information, see the ActiveMQ version management and the RabbitMQ version management sections in the Amazon MQ Developer Guide.

> **Type**: string
>
> **Required**: False

brokerId

The unique ID that Amazon MQ generates for the broker.

> **Type**: string
>
> **Required**: False

brokerArn

The broker's Amazon Resource Name (ARN).

> **Type**: string
>
> **Required**: False

deploymentMode

The broker's deployment mode.

Type: DeploymentMode

Required: True

maintenanceWindowStartTime

The parameters that determine the WeeklyStartTime.

Type: WeeklyStartTime

Required: False

created

The time when the broker was created.

Type: string

Required: False

Format: date-time

authenticationStrategy

The authentication strategy used to secure the broker. The default is SIMPLE.

Type: AuthenticationStrategy

Required: False

users

The list of all broker usernames for the specified broker.

Type: Array of type UserSummary

Required: False

tags

The list of all tags associated with this broker.

Type: object
Required: False

dataReplicationMetadata

The replication details of the data replication-enabled broker. Only returned if dataReplicationMode is set to CRDR.

Type: DataReplicationMetadataOutput
Required: False

pendingLdapServerMetadata

The metadata of the LDAP server that will be used to authenticate and authorize connections to the broker after it is rebooted.

Type: LdapServerMetadataOutput
Required: False

encryptionOptions

Encryption options for the broker.

Type: EncryptionOptions
Required: False

pendingDataReplicationMetadata

The pending replication details of the data replication-enabled broker. Only returned if pendingDataReplicationMode is set to CRDR.

Type: DataReplicationMetadataOutput
Required: False

storageType

The broker's storage type.

Type: BrokerStorageType

Required: False

actionsRequired

Actions required for a broker.

> **Type**: Array of type [ActionRequired](#)
> **Required**: False

securityGroups

The list of rules (1 minimum, 125 maximum) that authorize connections to brokers.

> **Type**: Array of type string
> **Required**: False

brokerName

The broker's name. This value must be unique in your AWS account account, 1-50 characters long, must contain only letters, numbers, dashes, and underscores, and must not contain white spaces, brackets, wildcard characters, or special characters.

> **Type**: string
> **Required**: False

autoMinorVersionUpgrade

Enables automatic upgrades to new patch versions for brokers as new versions are released and supported by Amazon MQ. Automatic upgrades occur during the scheduled maintenance window or after a manual broker reboot.

> **Type**: boolean
> **Required**: True

EncryptionOptions

Encryption options for the broker.

useAwsOwnedKey

Enables the use of an AWS owned CMK using AWS KMS (KMS). Set to `true` by default, if no value is provided, for example, for RabbitMQ brokers.

> **Type**: boolean
> **Required**: True

kmsKeyId

The customer master key (CMK) to use for the A AWS KMS (KMS). This key is used to encrypt your data at rest. If not provided, Amazon MQ will use a default CMK to encrypt your data.

> **Type**: string
> **Required**: False

EngineType

The type of broker engine. Amazon MQ supports ActiveMQ and RabbitMQ.

```
ACTIVEMQ
RABBITMQ
```

Error

Returns information about an error.

errorAttribute

The attribute which caused the error.

> **Type**: string
> **Required**: False

message

The explanation of the error.

> **Type**: string

Required: False

LdapServerMetadataInput

Optional. The metadata of the LDAP server used to authenticate and authorize connections to the broker.

> ⚠️ **Important**
>
> Does not apply to RabbitMQ brokers.

roleSearchMatching

The LDAP search filter used to find roles within the `roleBase`. The distinguished name of the user matched by `userSearchMatching` is substituted into the `{0}` placeholder in the search filter. The client's username is substituted into the `{1}` placeholder. For example, if you set this option to `(member=uid={1})`for the user `janedoe`, the search filter becomes `(member=uid=janedoe)` after string substitution. It matches all role entries that have a member attribute equal to `uid=janedoe` under the subtree selected by the `roleBase`.

   Type: string
   Required: True

serviceAccountPassword

Service account password. A service account is an account in your LDAP server that has access to initiate a connection. For example, `cn=admin,dc=corp, dc=example, dc=com`.

   Type: string
   Required: True

roleBase

The distinguished name of the node in the directory information tree (DIT) to search for roles or groups. For example, `ou=group, ou=corp, dc=corp, dc=example, dc=com`.

   Type: string
   Required: True

hosts

Specifies the location of the LDAP server such as AWS Directory Service for Microsoft Active Directory. Optional failover server.

Type: Array of type string

Required: True

roleName

Specifies the LDAP attribute that identifies the group name attribute in the object returned from the group membership query.

Type: string

Required: False

userBase

Select a particular subtree of the directory information tree (DIT) to search for user entries. The subtree is specified by a DN, which specifies the base node of the subtree. For example, by setting this option to ou=Users,ou=corp, dc=corp, dc=example, dc=com, the search for user entries is restricted to the subtree beneath ou=Users, ou=corp, dc=corp, dc=example, dc=com.

Type: string

Required: True

roleSearchSubtree

The directory search scope for the role. If set to true, scope is to search the entire subtree.

Type: boolean

Required: False

serviceAccountUsername

Service account username. A service account is an account in your LDAP server that has access to initiate a connection. For example, cn=admin,dc=corp, dc=example, dc=com.

Type: string
Required: True

userRoleName

Specifies the name of the LDAP attribute for the user group membership.

Type: string
Required: False

userSearchMatching

The LDAP search filter used to find users within the `userBase`. The client's username is substituted into the {0} placeholder in the search filter. For example, if this option is set to (`uid={0}`) and the received username is `janedoe`, the search filter becomes (`uid=janedoe`) after string substitution. It will result in matching an entry like `uid=janedoe, ou=Users,ou=corp, dc=corp, dc=example, dc=com`.

Type: string
Required: True

userSearchSubtree

The directory search scope for the user. If set to true, scope is to search the entire subtree.

Type: boolean
Required: False

LdapServerMetadataOutput

Optional. The metadata of the LDAP server used to authenticate and authorize connections to the broker.

roleSearchMatching

The LDAP search filter used to find roles within the `roleBase`. The distinguished name of the user matched by `userSearchMatching` is substituted into the {0} placeholder in the search filter. The client's username is substituted into the {1} placeholder. For example, if you set this option to (`member=uid={1}`)for the user `janedoe`, the search filter becomes (`member=uid=janedoe`)

after string substitution. It matches all role entries that have a member attribute equal to `uid=janedoe` under the subtree selected by the `roleBase`.

> **Type**: string
> **Required**: True

roleBase

The distinguished name of the node in the directory information tree (DIT) to search for roles or groups. For example, `ou=group, ou=corp, dc=corp, dc=example, dc=com`.

> **Type**: string
> **Required**: True

hosts

Specifies the location of the LDAP server such as AWS Directory Service for Microsoft Active Directory. Optional failover server.

> **Type**: Array of type string
> **Required**: True

roleName

Specifies the LDAP attribute that identifies the group name attribute in the object returned from the group membership query.

> **Type**: string
> **Required**: False

userBase

Select a particular subtree of the directory information tree (DIT) to search for user entries. The subtree is specified by a DN, which specifies the base node of the subtree. For example, by setting this option to `ou=Users,ou=corp, dc=corp, dc=example, dc=com`, the search for user entries is restricted to the subtree beneath `ou=Users, ou=corp, dc=corp, dc=example, dc=com`.

> **Type**: string

Required: True

roleSearchSubtree

The directory search scope for the role. If set to true, scope is to search the entire subtree.

   Type: boolean
   Required: False

serviceAccountUsername

Service account username. A service account is an account in your LDAP server that has access to initiate a connection. For example, `cn=admin,dc=corp, dc=example, dc=com`.

   Type: string
   Required: True

userRoleName

Specifies the name of the LDAP attribute for the user group membership.

   Type: string
   Required: False

userSearchMatching

The LDAP search filter used to find users within the `userBase`. The client's username is substituted into the {0} placeholder in the search filter. For example, if this option is set to `(uid={0})` and the received username is `janedoe`, the search filter becomes `(uid=janedoe)` after string substitution. It will result in matching an entry like `uid=janedoe, ou=Users,ou=corp, dc=corp, dc=example, dc=com`.

   Type: string
   Required: True

userSearchSubtree

The directory search scope for the user. If set to true, scope is to search the entire subtree.

Type: boolean
Required: False

Logs

The list of information about logs to be enabled for the specified broker.

general

Enables general logging.

    Type: boolean
    Required: False

audit

Enables audit logging. Every user management action made using JMX or the ActiveMQ Web Console is logged. Does not apply to RabbitMQ brokers.

    Type: boolean
    Required: False

LogsSummary

The list of information about logs currently enabled and pending to be deployed for the specified broker.

generalLogGroup

The location of the CloudWatch Logs log group where general logs are sent.

    Type: string
    Required: True

general

Enables general logging.

    Type: boolean

Required: True

audit

Enables audit logging. Every user management action made using JMX or the ActiveMQ Web Console is logged.

>   **Type**: boolean
>   **Required**: False

pending

The list of information about logs pending to be deployed for the specified broker.

>   **Type**: PendingLogs
>   **Required**: False

auditLogGroup

The location of the CloudWatch Logs log group where audit logs are sent.

>   **Type**: string
>   **Required**: False

PendingLogs

The list of information about logs to be enabled for the specified broker.

general

Enables general logging.

>   **Type**: boolean
>   **Required**: False

audit

Enables audit logging. Every user management action made using JMX or the ActiveMQ Web Console is logged.

Type: boolean
Required: False

UpdateBrokerInput

Updates the broker using the specified properties.

dataReplicationMode

Defines whether this broker is a part of a data replication pair.

Type: DataReplicationMode
Required: False

engineVersion

The broker engine version. For more information, see the ActiveMQ version management and the RabbitMQ version management sections in the Amazon MQ Developer Guide.

> ⓘ **Note**
>
> When upgrading to ActiveMQ version 5.18 and above or RabbitMQ version 3.13 and above, you must have `autoMinorVersionUpgrade` set to `true` for the broker.

Type: string
Required: False

maintenanceWindowStartTime

The parameters that determine the WeeklyStartTime.

Type: WeeklyStartTime
Required: False

configuration

A list of information about the configuration.

Type: ConfigurationId

Required: False

authenticationStrategy

Optional. The authentication strategy used to secure the broker. The default is SIMPLE.

Type: AuthenticationStrategy

Required: False

securityGroups

The list of security groups (1 minimum, 5 maximum) that authorizes connections to brokers.

Type: Array of type string

Required: False

ldapServerMetadata

Optional. The metadata of the LDAP server used to authenticate and authorize connections to the broker. Does not apply to RabbitMQ brokers.

Type: LdapServerMetadataInput

Required: False

logs

Enables Amazon CloudWatch logging for brokers.

Type: Logs

Required: False

hostInstanceType

The broker's host instance type to upgrade to. For a list of supported instance types, see Broker instance types.

Type: string

Required: False

autoMinorVersionUpgrade

Enables automatic upgrades to new patch versions for brokers as new versions are released and supported by Amazon MQ. Automatic upgrades occur during the scheduled maintenance window or after a manual broker reboot.

> ⓘ **Note**
>
> Must be set to `true` for ActiveMQ brokers version 5.18 and above and for RabbitMQ brokers version 3.13 and above.

  Type: boolean
  Required: False

UpdateBrokerOutput

Returns information about the updated broker.

engineVersion

The broker engine version to upgrade to. For more information, see the ActiveMQ version management and the RabbitMQ version management sections in the Amazon MQ Developer Guide.

  Type: string
  Required: False

brokerId

Required. The unique ID that Amazon MQ generates for the broker.

  Type: string
  Required: True

maintenanceWindowStartTime

The parameters that determine the WeeklyStartTime.

Type: WeeklyStartTime

Required: False

configuration

The ID of the updated configuration.

Type: ConfigurationId

Required: False

authenticationStrategy

Optional. The authentication strategy used to secure the broker. The default is SIMPLE.

Type: AuthenticationStrategy

Required: False

pendingDataReplicationMode

Describes whether this broker will be a part of a data replication pair after reboot.

Type: DataReplicationMode

Required: False

hostInstanceType

The broker's host instance type to upgrade to. For a list of supported instance types, see Broker instance types.

Type: string

Required: False

dataReplicationMode

Describes whether this broker is a part of a data replication pair.

Type: DataReplicationMode

Required: False

dataReplicationMetadata

The replication details of the data replication-enabled broker. Only returned if
dataReplicationMode is set to CRDR.

   Type: DataReplicationMetadataOutput
   Required: False

pendingDataReplicationMetadata

The pending replication details of the data replication-enabled broker. Only returned if
pendingDataReplicationMode is set to CRDR.

   Type: DataReplicationMetadataOutput
   Required: False

securityGroups

The list of security groups (1 minimum, 5 maximum) that authorizes connections to brokers.

   Type: Array of type string
   Required: False

ldapServerMetadata

Optional. The metadata of the LDAP server used to authenticate and authorize connections to the
broker. Does not apply to RabbitMQ brokers.

   Type: LdapServerMetadataOutput
   Required: False

logs

The list of information about logs to be enabled for the specified broker.

   Type: Logs
   Required: False

autoMinorVersionUpgrade

Enables automatic upgrades to new patch versions for brokers as new versions are released and supported by Amazon MQ. Automatic upgrades occur during the scheduled maintenance window or after a manual broker reboot.

> **Type**: boolean
> **Required**: False

UserSummary

Returns a list of all broker users. Does not apply to RabbitMQ brokers.

pendingChange

The type of change pending for the broker user.

> **Type**: ChangeType
> **Required**: False

username

Required. The username of the broker user. This value can contain only alphanumeric characters, dashes, periods, underscores, and tildes (- . _ ~). This value must be 2-100 characters long.

> **Type**: string
> **Required**: True

WeeklyStartTime

The scheduled time period relative to UTC during which Amazon MQ begins to apply pending updates or patches to the broker.

dayOfWeek

Required. The day of the week.

> **Type**: string
> **Required**: True

Values: MONDAY | TUESDAY | WEDNESDAY | THURSDAY | FRIDAY | SATURDAY | SUNDAY

timeZone

The time zone, UTC by default, in either the Country/City format, or the UTC offset format.

Type: string

Required: False

timeOfDay

Required. The time, in 24-hour format.

Type: string

Required: True

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

DescribeBroker

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

UpdateBroker

- [AWS Command Line Interface](#)

- [AWS SDK for .NET](#)

- [AWS SDK for C++](#)

- [AWS SDK for Go v2](#)

- [AWS SDK for Java V2](#)

- [AWS SDK for JavaScript V3](#)

- [AWS SDK for PHP V3](#)

- [AWS SDK for Python](#)

- [AWS SDK for Ruby V3](#)

DeleteBroker

- [AWS Command Line Interface](#)

- [AWS SDK for .NET](#)

- [AWS SDK for C++](#)

- [AWS SDK for Go v2](#)

- [AWS SDK for Java V2](#)

- [AWS SDK for JavaScript V3](#)

- [AWS SDK for PHP V3](#)

- [AWS SDK for Python](#)

- [AWS SDK for Ruby V3](#)

Broker Engine Types

Retrieve information about available broker engines. AWS does not support all instance types in all availability zones and regions. For more information, see the [ActiveMQ version management](#) and the [RabbitMQ version management](#) sections in the Amazon MQ Developer Guide.

This API will tell you, for a given region and availability zone, which broker engine types and engine versions you can create.

URI

`/v1/broker-engine-types`

HTTP methods

GET

Operation ID: `DescribeBrokerEngineTypes`

Describe available engine types and versions.

Query parameters

Name	Type	Required	Description
engineType	String	False	Filter response by engine type.
nextToken	String	False	The token that specifies the next page of results Amazon MQ should return. To request the first page, leave nextToken empty.
maxResults	String	False	The maximum number of brokers that Amazon MQ can return per page (20 by default). This value must be an integer from 5 to 100.

Responses

Status code	Response model	Description
200	[BrokerEngineTypeOutput](#)	HTTP Status Code 200: OK.
400	[Error](#)	HTTP Status Code 400: Bad request due to incorrect input. Correct your request and then retry it.
403	[Error](#)	HTTP Status Code 403: Access forbidden. Correct your credentials and then retry your request.
500	[Error](#)	HTTP Status Code 500: Unexpected internal server error. Retrying your request might resolve the issue.

OPTIONS

Responses

Status code	Response model	Description
200	None	Default response for CORS method

Schemas

Response bodies

BrokerEngineTypeOutput schema

```
{
```

```
    "nextToken": "string",
    "maxResults": integer,
    "brokerEngineTypes": [
      {
        "engineVersions": [
          {
            "name": "string"
          }
        ],
        "engineType": enum
      }
    ]
 }
```

Error schema

```
{
   "errorAttribute": "string",
   "message": "string"
}
```

Properties

BrokerEngineType

Types of broker engines.

engineVersions

The list of engine versions.

   Type: Array of type EngineVersion
   Required: False

engineType

The broker's engine type.

   Type: EngineType
   Required: False

BrokerEngineTypeOutput

Returns a list of broker engine type.

nextToken

The token that specifies the next page of results Amazon MQ should return. To request the first page, leave nextToken empty.

Type: string
Required: False

maxResults

Required. The maximum number of engine types that can be returned per page (20 by default). This value must be an integer from 5 to 100.

Type: integer
Required: True
Minimum: 5
Maximum: 100

brokerEngineTypes

List of available engine types and versions.

Type: Array of type BrokerEngineType
Required: False

EngineType

The type of broker engine. Amazon MQ supports ActiveMQ and RabbitMQ.

```
ACTIVEMQ
RABBITMQ
```

EngineVersion

Id of the engine version.

name

Id for the version.

> **Type**: string
> **Required**: False

Error

Returns information about an error.

errorAttribute

The attribute which caused the error.

> **Type**: string
> **Required**: False

message

The explanation of the error.

> **Type**: string
> **Required**: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

DescribeBrokerEngineTypes

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2

- [AWS SDK for JavaScript V3](#)

- [AWS SDK for PHP V3](#)

- [AWS SDK for Python](#)

- [AWS SDK for Ruby V3](#)

Broker Instance Options

You can retrieve information about broker instances. For more information about the different components of an Amazon MQ broker, see [How Amazon MQ works](#) in the *Amazon MQ Developer Guide*.

URI

`/v1/broker-instance-options`

HTTP methods

GET

Operation ID: `DescribeBrokerInstanceOptions`

Describe available broker instance options.

Query parameters

Name	Type	Required	Description
hostInstanceType	String	False	Filter response by host instance type.
nextToken	String	False	The token that specifies the next page of results Amazon MQ should return. To request the first page, leave nextToken empty.

Name	Type	Required	Description
storageType	String	False	Filter response by storage type.
maxResults	String	False	The maximum number of brokers that Amazon MQ can return per page (20 by default). This value must be an integer from 5 to 100.
engineType	String	False	Filter response by engine type.

Responses

Status code	Response model	Description
200	BrokerInstanceOptionsOutput	HTTP Status Code 200: OK.
400	Error	HTTP Status Code 400: Bad request due to incorrect input. Correct your request and then retry it.
403	Error	HTTP Status Code 403: Access forbidden. Correct your credentials and then retry your request.
500	Error	HTTP Status Code 500: Unexpected internal server error. Retrying your request might resolve the issue.

OPTIONS

Responses

Status code	Response model	Description
200	None	Default response for CORS method

Schemas

Response bodies

BrokerInstanceOptionsOutput schema

```
{
  "nextToken": "string",
  "maxResults": integer,
  "brokerInstanceOptions": [
    {
      "supportedDeploymentModes": [
        enum
      ],
      "supportedEngineVersions": [
        "string"
      ],
      "storageType": enum,
      "engineType": enum,
      "availabilityZones": [
        {
          "name": "string"
        }
      ],
      "hostInstanceType": "string"
    }
  ]
}
```

Error schema

```
{
```

```
    "errorAttribute": "string",
    "message": "string"
}
```

Properties

AvailabilityZone

Name of the availability zone.

name

Id for the availability zone.

> **Type**: string
> **Required**: False

BrokerInstanceOption

Option for host instance type.

supportedDeploymentModes

The list of supported deployment modes.

> **Type**: Array of type [DeploymentMode](#)
> **Required**: False

supportedEngineVersions

The list of supported engine versions.

> **Type**: Array of type string
> **Required**: False

storageType

The broker's storage type.

> **Type**: [BrokerStorageType](#)

Required: False

engineType

The broker's engine type.

> **Type**: EngineType
> **Required**: False

availabilityZones

The list of available az.

> **Type**: Array of type AvailabilityZone
> **Required**: False

hostInstanceType

The broker's instance type.

> **Type**: string
> **Required**: False

BrokerInstanceOptionsOutput

Returns a list of broker instance options.

nextToken

The token that specifies the next page of results Amazon MQ should return. To request the first page, leave nextToken empty.

> **Type**: string
> **Required**: False

maxResults

Required. The maximum number of instance options that can be returned per page (20 by default). This value must be an integer from 5 to 100.

Type: integer

Required: True

Minimum: 5

Maximum: 100

brokerInstanceOptions

List of available broker instance options.

Type: Array of type [BrokerInstanceOption](#)

Required: False

BrokerStorageType

The broker's storage type.

> ⚠️ **Important**
>
> EFS is not supported for RabbitMQ engine type.

EBS

EFS

DeploymentMode

The broker's deployment mode.

```
SINGLE_INSTANCE
ACTIVE_STANDBY_MULTI_AZ
CLUSTER_MULTI_AZ
```

EngineType

The type of broker engine. Amazon MQ supports ActiveMQ and RabbitMQ.

```
ACTIVEMQ
```

```
RABBITMQ
```

Error

Returns information about an error.

errorAttribute

The attribute which caused the error.

Type: string
Required: False

message

The explanation of the error.

Type: string
Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

DescribeBrokerInstanceOptions

- AWS Command Line Interface
- [AWS SDK for .NET](AWS SDK for .NET)
- [AWS SDK for C++](AWS SDK for C++)
- [AWS SDK for Go v2](AWS SDK for Go v2)
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- [AWS SDK for PHP V3](AWS SDK for PHP V3)
- AWS SDK for Python
- AWS SDK for Ruby V3

Broker Reboot

To apply a new configuration to a broker, you can reboot the broker. In addition, if your broker becomes unresponsive, you can reboot it to recover from a faulty state.

> **ⓘ Note**
>
> You can reboot only a broker with the RUNNING status.

URI

/v1/brokers/*broker-id*/reboot

HTTP methods

POST

Operation ID: RebootBroker

Reboots a broker. Note: This API is asynchronous.

Path parameters

Name	Type	Required	Description
broker-id	String	True	The unique ID that Amazon MQ generates for the broker.

Responses

Status code	Response model	Description
200	None	HTTP Status Code 200: OK.
400	Error	HTTP Status Code 400: Bad request due to incorrect

Status code	Response model	Description
		input. Correct your request and then retry it.
403	Error	HTTP Status Code 403: Access forbidden. Correct your credentials and then retry your request.
404	Error	HTTP Status Code 404: Resource not found due to incorrect input. Correct your request and then retry it.
500	Error	HTTP Status Code 500: Unexpected internal server error. Retrying your request might resolve the issue.

OPTIONS

Path parameters

Name	Type	Required	Description
broker-id	String	True	The unique ID that Amazon MQ generates for the broker.

Responses

Status code	Response model	Description
200	None	Default response for CORS method

Schemas

Response bodies

Error schema

```
{
  "errorAttribute": "string",
  "message": "string"
}
```

Properties

Error

Returns information about an error.

errorAttribute

The attribute which caused the error.

>**Type**: string
>**Required**: False

message

The explanation of the error.

>**Type**: string
>**Required**: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

RebootBroker

- AWS Command Line Interface

- [AWS SDK for .NET](#)

- [AWS SDK for C++](#)

- [AWS SDK for Go v2](#)

- [AWS SDK for Java V2](#)

- [AWS SDK for JavaScript V3](#)

- [AWS SDK for PHP V3](#)

- [AWS SDK for Python](#)

- [AWS SDK for Ruby V3](#)

Brokers

This is a collection of brokers. A broker is a message broker environment running on Amazon MQ. It is the basic building block of Amazon MQ. For more information, see [Broker instance types](#) in the *Amazon MQ Developer Guide*.

URI

`/v1/brokers`

HTTP methods

GET

Operation ID: `ListBrokers`

Returns a list of all brokers.

Query parameters

Name	Type	Required	Description
nextToken	String	False	The token that specifies the next page of results Amazon MQ should return. To request

Name	Type	Required	Description
			the first page, leave nextToken empty.
maxResults	String	False	The maximum number of brokers that Amazon MQ can return per page (20 by default). This value must be an integer from 5 to 100.

Responses

Status code	Response model	Description
200	ListBrokersOutput	HTTP Status Code 200: OK.
400	Error	HTTP Status Code 400: Bad request due to incorrect input. Correct your request and then retry it.
403	Error	HTTP Status Code 403: Access forbidden. Correct your credentials and then retry your request.
500	Error	HTTP Status Code 500: Unexpected internal server error. Retrying your request might resolve the issue.

POST

Operation ID: CreateBroker

Creates a broker. Note: This API is asynchronous.

To create a broker, you must either use the `AmazonMQFullAccess` IAM policy or include the following EC2 permissions in your IAM policy.

- `ec2:CreateNetworkInterface`

  This permission is required to allow Amazon MQ to create an elastic network interface (ENI) on behalf of your account.

- `ec2:CreateNetworkInterfacePermission`

  This permission is required to attach the ENI to the broker instance.

- `ec2:DeleteNetworkInterface`

- `ec2:DeleteNetworkInterfacePermission`

- `ec2:DetachNetworkInterface`

- `ec2:DescribeInternetGateways`

- `ec2:DescribeNetworkInterfaces`

- `ec2:DescribeNetworkInterfacePermissions`

- `ec2:DescribeRouteTables`

- `ec2:DescribeSecurityGroups`

- `ec2:DescribeSubnets`

- `ec2:DescribeVpcs`

For more information, see [Create an IAM User and Get Your AWS Credentials](#) and [Never Modify or Delete the Amazon MQ Elastic Network Interface](#) in the *Amazon MQ Developer Guide*.

Responses

Status code	Response model	Description
200	[CreateBrokerOutput](#)	HTTP Status Code 200: OK.
400	[Error](#)	HTTP Status Code 400: Bad request due to incorrect input. Correct your request and then retry it.

Status code	Response model	Description
401	[Error](#)	HTTP Status Code 401: Unauthorized request. The provided credentials couldn't be validated.
403	[Error](#)	HTTP Status Code 403: Access forbidden. Correct your credentials and then retry your request.
409	[Error](#)	HTTP Status Code 409: Conflict. This broker name already exists. Retry your request with another name.
500	[Error](#)	HTTP Status Code 500: Unexpected internal server error. Retrying your request might resolve the issue.

OPTIONS

Responses

Status code	Response model	Description
200	None	Default response for CORS method

Schemas

Request bodies

POST schema

```
{
```

```
  "engineVersion": "string",
  "deploymentMode": enum,
  "maintenanceWindowStartTime": {
    "dayOfWeek": enum,
    "timeZone": "string",
    "timeOfDay": "string"
  },
  "configuration": {
    "id": "string",
    "revision": integer
  },
  "authenticationStrategy": enum,
  "engineType": enum,
  "hostInstanceType": "string",
  "users": [
    {
      "password": "string",
      "replicationUser": boolean,
      "groups": [
        "string"
      ],
      "consoleAccess": boolean,
      "username": "string"
    }
  ],
  "tags": {
  },
  "dataReplicationMode": enum,
  "creatorRequestId": "string",
  "publiclyAccessible": boolean,
  "encryptionOptions": {
    "useAwsOwnedKey": boolean,
    "kmsKeyId": "string"
  },
  "storageType": enum,
  "dataReplicationPrimaryBrokerArn": "string",
  "securityGroups": [
    "string"
  ],
  "brokerName": "string",
  "ldapServerMetadata": {
    "roleSearchMatching": "string",
    "serviceAccountPassword": "string",
    "roleBase": "string",
```

```
      "hosts": [
        "string"
      ],
      "roleName": "string",
      "userBase": "string",
      "roleSearchSubtree": boolean,
      "serviceAccountUsername": "string",
      "userRoleName": "string",
      "userSearchMatching": "string",
      "userSearchSubtree": boolean
    },
    "logs": {
      "general": boolean,
      "audit": boolean
    },
    "autoMinorVersionUpgrade": boolean,
    "subnetIds": [
      "string"
    ]
  }
```

Response bodies

ListBrokersOutput schema

```
{
  "brokerSummaries": [
    {
      "brokerArn": "string",
      "brokerId": "string",
      "deploymentMode": enum,
      "created": "string",
      "brokerState": enum,
      "engineType": enum,
      "brokerName": "string",
      "hostInstanceType": "string"
    }
  ],
  "nextToken": "string"
}
```

CreateBrokerOutput schema

```
{
  "brokerArn": "string",
  "brokerId": "string"
}
```

Error schema

```
{
  "errorAttribute": "string",
  "message": "string"
}
```

Properties

AuthenticationStrategy

Optional. The authentication strategy used to secure the broker. The default is SIMPLE.

```
SIMPLE
LDAP
```

BrokerState

The broker's status.

```
CREATION_IN_PROGRESS
CREATION_FAILED
DELETION_IN_PROGRESS
RUNNING
REBOOT_IN_PROGRESS
CRITICAL_ACTION_REQUIRED
REPLICA
```

BrokerStorageType

The broker's storage type.

> ⚠ **Important**
>
> EFS is not supported for RabbitMQ engine type.

EBS
EFS

BrokerSummary

Returns information about all brokers.

brokerArn

The broker's Amazon Resource Name (ARN).

Type: string
Required: False

brokerId

The unique ID that Amazon MQ generates for the broker.

Type: string
Required: False

deploymentMode

The broker's deployment mode.

Type: DeploymentMode
Required: True

created

The time when the broker was created.

Type: string
Required: False

Format: date-time

brokerState

The broker's status.

> **Type**: BrokerState
> **Required**: False

engineType

The type of broker engine.

> **Type**: EngineType
> **Required**: True

brokerName

The broker's name. This value is unique in your AWS account, 1-50 characters long, and containing only letters, numbers, dashes, and underscores, and must not contain white spaces, brackets, wildcard characters, or special characters.

> **Type**: string
> **Required**: False

hostInstanceType

The broker's instance type.

> **Type**: string
> **Required**: False

ConfigurationId

A list of information about the configuration.

id

Required. The unique ID that Amazon MQ generates for the configuration.

Type: string

Required: True

revision

The revision number of the configuration.

Type: integer

Required: False

CreateBrokerInput

Creates a broker.

engineVersion

The broker engine version. Defaults to the latest available version for the specified broker engine type. For more information, see the [ActiveMQ version management](#) and the [RabbitMQ version management](#) sections in the Amazon MQ Developer Guide.

Type: string

Required: False

deploymentMode

Required. The broker's deployment mode.

Type: [DeploymentMode](#)

Required: True

maintenanceWindowStartTime

The parameters that determine the WeeklyStartTime.

Type: [WeeklyStartTime](#)

Required: False

configuration

A list of information about the configuration.

Type: ConfigurationId
Required: False

authenticationStrategy

Optional. The authentication strategy used to secure the broker. The default is SIMPLE.

Type: AuthenticationStrategy
Required: False

engineType

Required. The type of broker engine. Currently, Amazon MQ supports ACTIVEMQ and RABBITMQ.

Type: EngineType
Required: True

hostInstanceType

Required. The broker's instance type.

Type: string
Required: True

users

The list of broker users (persons or applications) who can access queues and topics. For Amazon MQ for RabbitMQ brokers, one and only one administrative user is accepted and created when a broker is first provisioned. All subsequent broker users are created by making RabbitMQ API calls directly to brokers or via the RabbitMQ web console.

Type: Array of type User
Required: True

tags

Create tags when creating the broker.

> **Type**: object
> **Required**: False

dataReplicationMode

Defines whether this broker is a part of a data replication pair.

> **Type**: DataReplicationMode
> **Required**: False

creatorRequestId

The unique ID that the requester receives for the created broker. Amazon MQ passes your ID with the API action.

> ⓘ **Note**
>
> We recommend using a Universally Unique Identifier (UUID) for the creatorRequestId. You may omit the `creatorRequestId` if your application doesn't require idempotency.

> **Type**: string
> **Required**: False

publiclyAccessible

Enables connections from applications outside of the VPC that hosts the broker's subnets. Set to `false` by default, if no value is provided.

> **Type**: boolean
> **Required**: True

encryptionOptions

Encryption options for the broker.

Type: [EncryptionOptions](#)

Required: False

storageType

The broker's storage type.

Type: [BrokerStorageType](#)

Required: False

dataReplicationPrimaryBrokerArn

The Amazon Resource Name (ARN) of the primary broker that is used to replicate data from in a data replication pair, and is applied to the replica broker. Must be set when dataReplicationMode is set to CRDR.

Type: string

Required: False

securityGroups

The list of rules (1 minimum, 125 maximum) that authorize connections to brokers.

Type: Array of type string

Required: False

brokerName

Required. The broker's name. This value must be unique in your AWS account, 1-50 characters long, must contain only letters, numbers, dashes, and underscores, and must not contain white spaces, brackets, wildcard characters, or special characters.

> ⚠️ **Important**
>
> Do not add personally identifiable information (PII) or other confidential or sensitive information in broker names. Broker names are accessible to other AWS services, including CloudWatch Logs. Broker names are not intended to be used for private or sensitive data.

Type: string

Required: True

ldapServerMetadata

Optional. The metadata of the LDAP server used to authenticate and authorize connections to the broker. Does not apply to RabbitMQ brokers.

> **Type**: LdapServerMetadataInput
> **Required**: False

logs

Enables Amazon CloudWatch logging for brokers.

> **Type**: Logs
> **Required**: False

autoMinorVersionUpgrade

Enables automatic upgrades to new patch versions for brokers as new versions are released and supported by Amazon MQ. Automatic upgrades occur during the scheduled maintenance window or after a manual broker reboot. Set to `true` by default, if no value is specified.

> ⓘ **Note**
>
> Must be set to `true` for ActiveMQ brokers version 5.18 and above and for RabbitMQ brokers version 3.13 and above.

> **Type**: boolean
> **Required**: False

subnetIds

The list of groups that define which subnets and IP ranges the broker can use from different Availability Zones. If you specify more than one subnet, the subnets must be in different

Availability Zones. Amazon MQ will not be able to create VPC endpoints for your broker with multiple subnets in the same Availability Zone. A SINGLE_INSTANCE deployment requires one subnet (for example, the default subnet). An ACTIVE_STANDBY_MULTI_AZ Amazon MQ for ActiveMQ deployment requires two subnets. A CLUSTER_MULTI_AZ Amazon MQ for RabbitMQ deployment has no subnet requirements when deployed with public accessibility. Deployment without public accessibility requires at least one subnet.

> ⚠️ **Important**
>
> If you specify subnets in a [shared VPC](#) for a RabbitMQ broker, the associated VPC to which the specified subnets belong must be owned by your AWS account. Amazon MQ will not be able to create VPC endpoints in VPCs that are not owned by your AWS account.

Type: Array of type string
Required: False

CreateBrokerOutput

Returns information about the created broker.

brokerArn

The broker's Amazon Resource Name (ARN).

Type: string
Required: False

brokerId

The unique ID that Amazon MQ generates for the broker.

Type: string
Required: False

DataReplicationMode

Specifies whether a broker is a part of a data replication pair.

```
NONE
CRDR
```

DeploymentMode

The broker's deployment mode.

```
SINGLE_INSTANCE
ACTIVE_STANDBY_MULTI_AZ
CLUSTER_MULTI_AZ
```

EncryptionOptions

Encryption options for the broker.

useAwsOwnedKey

Enables the use of an AWS owned CMK using AWS KMS (KMS). Set to `true` by default, if no value is provided, for example, for RabbitMQ brokers.

Type: boolean
Required: True

kmsKeyId

The customer master key (CMK) to use for the A AWS KMS (KMS). This key is used to encrypt your data at rest. If not provided, Amazon MQ will use a default CMK to encrypt your data.

Type: string
Required: False

EngineType

The type of broker engine. Amazon MQ supports ActiveMQ and RabbitMQ.

```
ACTIVEMQ
RABBITMQ
```

Error

Returns information about an error.

errorAttribute

The attribute which caused the error.

>   **Type**: string
>   **Required**: False

message

The explanation of the error.

>   **Type**: string
>   **Required**: False

LdapServerMetadataInput

Optional. The metadata of the LDAP server used to authenticate and authorize connections to the broker.

> ⚠️ **Important**
>
>   Does not apply to RabbitMQ brokers.

roleSearchMatching

The LDAP search filter used to find roles within the `roleBase`. The distinguished name of the user matched by `userSearchMatching` is substituted into the `{0}` placeholder in the search filter. The client's username is substituted into the `{1}` placeholder. For example, if you set this option to `(member=uid={1})`for the user `janedoe`, the search filter becomes `(member=uid=janedoe)` after string substitution. It matches all role entries that have a member attribute equal to `uid=janedoe` under the subtree selected by the `roleBase`.

>   **Type**: string

Required: True

serviceAccountPassword

Service account password. A service account is an account in your LDAP server that has access to initiate a connection. For example, `cn=admin,dc=corp, dc=example, dc=com`.

   Type: string
   Required: True

roleBase

The distinguished name of the node in the directory information tree (DIT) to search for roles or groups. For example, `ou=group, ou=corp, dc=corp, dc=example, dc=com`.

   Type: string
   Required: True

hosts

Specifies the location of the LDAP server such as AWS Directory Service for Microsoft Active Directory. Optional failover server.

   Type: Array of type string
   Required: True

roleName

Specifies the LDAP attribute that identifies the group name attribute in the object returned from the group membership query.

   Type: string
   Required: False

userBase

Select a particular subtree of the directory information tree (DIT) to search for user entries. The subtree is specified by a DN, which specifies the base node of the subtree. For example, by setting

this option to ou=Users,ou=corp, dc=corp, dc=example, dc=com, the search for user entries is restricted to the subtree beneath ou=Users, ou=corp, dc=corp, dc=example, dc=com.

 Type: string
 Required: True

roleSearchSubtree

The directory search scope for the role. If set to true, scope is to search the entire subtree.

 Type: boolean
 Required: False

serviceAccountUsername

Service account username. A service account is an account in your LDAP server that has access to initiate a connection. For example, cn=admin,dc=corp, dc=example, dc=com.

 Type: string
 Required: True

userRoleName

Specifies the name of the LDAP attribute for the user group membership.

 Type: string
 Required: False

userSearchMatching

The LDAP search filter used to find users within the userBase. The client's username is substituted into the {0} placeholder in the search filter. For example, if this option is set to (uid={0}) and the received username is janedoe, the search filter becomes (uid=janedoe) after string substitution. It will result in matching an entry like uid=janedoe, ou=Users,ou=corp, dc=corp, dc=example, dc=com.

 Type: string
 Required: True

userSearchSubtree

The directory search scope for the user. If set to true, scope is to search the entire subtree.

> **Type**: boolean
> **Required**: False

ListBrokersOutput

brokerSummaries

A list of information about all brokers.

> **Type**: Array of type [BrokerSummary](#)
> **Required**: False

nextToken

The token that specifies the next page of results Amazon MQ should return. To request the first page, leave nextToken empty.

> **Type**: string
> **Required**: False

Logs

The list of information about logs to be enabled for the specified broker.

general

Enables general logging.

> **Type**: boolean
> **Required**: False

audit

Enables audit logging. Every user management action made using JMX or the ActiveMQ Web Console is logged. Does not apply to RabbitMQ brokers.

Type: boolean
Required: False

User

A user associated with the broker. For Amazon MQ for RabbitMQ brokers, one and only one administrative user is accepted and created when a broker is first provisioned. All subsequent broker users are created by making RabbitMQ API calls directly to brokers or via the RabbitMQ web console.

password

Required. The password of the user. This value must be at least 12 characters long, must contain at least 4 unique characters, and must not contain commas, colons, or equal signs (,:=).

Type: string
Required: True
Format: password

replicationUser

Defines if this user is intended for CRDR replication purposes.

Type: boolean
Required: False

groups

The list of groups (20 maximum) to which the ActiveMQ user belongs. This value can contain only alphanumeric characters, dashes, periods, underscores, and tildes (- . _ ~). This value must be 2-100 characters long. Does not apply to RabbitMQ brokers.

Type: Array of type string
Required: False

consoleAccess

Enables access to the ActiveMQ Web Console for the ActiveMQ user. Does not apply to RabbitMQ brokers.

Type: boolean
Required: False

username

The username of the broker user. The following restrictions apply to broker usernames:

- For Amazon MQ for ActiveMQ brokers, this value can contain only alphanumeric characters, dashes, periods, underscores, and tildes (- . _ ~). This value must be 2-100 characters long.

- For Amazon MQ for RabbitMQ brokers, this value can contain only alphanumeric characters, dashes, periods, underscores (- . _). This value must not contain a tilde (~) character. Amazon MQ prohibts using `guest` as a valid usename. This value must be 2-100 characters long.

> ⚠️ **Important**
>
> Do not add personally identifiable information (PII) or other confidential or sensitive information in broker usernames. Broker usernames are accessible to other AWS services, including CloudWatch Logs. Broker usernames are not intended to be used for private or sensitive data.

Type: string
Required: True

WeeklyStartTime

The scheduled time period relative to UTC during which Amazon MQ begins to apply pending updates or patches to the broker.

dayOfWeek

Required. The day of the week.

Type: string
Required: True
Values: MONDAY | TUESDAY | WEDNESDAY | THURSDAY | FRIDAY | SATURDAY | SUNDAY

timeZone

The time zone, UTC by default, in either the Country/City format, or the UTC offset format.

> **Type**: string
> **Required**: False

timeOfDay

Required. The time, in 24-hour format.

> **Type**: string
> **Required**: True

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

ListBrokers

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

CreateBroker

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)

- [AWS SDK for Go v2](#)

- [AWS SDK for Java V2](#)

- [AWS SDK for JavaScript V3](#)

- [AWS SDK for PHP V3](#)

- [AWS SDK for Python](#)

- [AWS SDK for Ruby V3](#)

Brokers broker-id Promote

URI

/v1/brokers/*broker-id*/promote

HTTP methods

POST

Operation ID: `Promote`

Promotes a data replication replica broker to a primary.

Path parameters

Name	Type	Required	Description
broker-id	String	True	The unique ID that Amazon MQ generates for the broker.

Responses

Status code	Response model	Description
200	[PromoteOutput](#)	HTTP Status Code 200: OK.
400	[Error](#)	HTTP Status Code 400: Bad request due to incorrect

Status code	Response model	Description
		input. Correct your request and then retry it.
403	Error	HTTP Status Code 403: Access forbidden. Correct your credentials and then retry your request.
404	Error	HTTP Status Code 404: Resource not found due to incorrect input. Correct your request and then retry it.
500	Error	HTTP Status Code 500: Unexpected internal server error. Retrying your request might resolve the issue.

OPTIONS

Path parameters

Name	Type	Required	Description
broker-id	String	True	The unique ID that Amazon MQ generates for the broker.

Responses

Status code	Response model	Description
200	None	Default response for CORS method

Schemas

Request bodies

POST schema

```
{
    "mode": enum
}
```

Response bodies

PromoteOutput schema

```
{
    "brokerId": "string"
}
```

Error schema

```
{
    "errorAttribute": "string",
    "message": "string"
}
```

Properties

Error

Returns information about an error.

errorAttribute

The attribute which caused the error.

> **Type**: string
> **Required**: False

message

The explanation of the error.

Type: string
Required: False

PromoteInput

Creates a Promote request with the properties specified.

mode

The Promote mode requested. Note: Valid values for the parameter are SWITCHOVER, FAILOVER.

Type: [PromoteMode](#)
Required: True

PromoteMode

The Promote mode requested.

```
SWITCHOVER
FAILOVER
```

PromoteOutput

Returns information about the updated broker.

brokerId

The unique ID that Amazon MQ generates for the broker.

Type: string
Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

Promote

- [AWS Command Line Interface](#)

- [AWS SDK for .NET](#)

- [AWS SDK for C++](#)

- [AWS SDK for Go v2](#)

- [AWS SDK for Java V2](#)

- [AWS SDK for JavaScript V3](#)

- [AWS SDK for PHP V3](#)

- [AWS SDK for Python](#)

- [AWS SDK for Ruby V3](#)

Configuration

A configuration contains all of the settings for your broker. For more information, see
[Configuration](#) and [Amazon MQ Broker Configuration Parameters](#) in the Amazon MQ Developer
Guide.

You can create a configuration before creating any brokers. You can then apply the configuration to
one or more brokers.

> ⚠️ **Important**
>
> Making changes to a configuration does not apply the changes to the broker immediately.
> To apply your changes, you must wait for the next maintenance window or reboot the
> broker.

URI

/v1/configurations/*configuration-id*

HTTP methods

GET

Operation ID: `DescribeConfiguration`

Returns information about the specified configuration.

Path parameters

Name	Type	Required	Description
configuration-id	String	True	The unique ID that Amazon MQ generates for the configuration.

Responses

Status code	Response model	Description
200	Configuration	HTTP Status Code 200: OK.
400	Error	HTTP Status Code 400: Bad request due to incorrect input. Correct your request and then retry it.
403	Error	HTTP Status Code 403: Access forbidden. Correct your credentials and then retry your request.
404	Error	HTTP Status Code 404: Resource not found due to incorrect input. Correct your request and then retry it.
500	Error	HTTP Status Code 500: Unexpected internal server error. Retrying your request might resolve the issue.

PUT

Operation ID: UpdateConfiguration

Updates the specified configuration.

Path parameters

Name	Type	Required	Description
configuration-id	String	True	The unique ID that Amazon MQ generates for the configuration.

Responses

Status code	Response model	Description
200	UpdateConfiguratio nOutput	HTTP Status Code 200: OK.
400	Error	HTTP Status Code 400: Bad request due to incorrect input. Correct your request and then retry it.
403	Error	HTTP Status Code 403: Access forbidden. Correct your credentials and then retry your request.
404	Error	HTTP Status Code 404: Resource not found due to incorrect input. Correct your request and then retry it.
409	Error	HTTP Status Code 409: Conflict. This broker name already exists. Retry your request with another name.

Status code	Response model	Description
500	[Error](#)	HTTP Status Code 500: Unexpected internal server error. Retrying your request might resolve the issue.

OPTIONS

Path parameters

Name	Type	Required	Description
configuration-id	String	True	The unique ID that Amazon MQ generates for the configuration.

Responses

Status code	Response model	Description
200	None	Default response for CORS method

Schemas

Request bodies

PUT schema

```
{
  "data": "string",
  "description": "string"
}
```

Response bodies

Configuration schema

```
{
  "engineVersion": "string",
  "created": "string",
  "authenticationStrategy": enum,
  "name": "string",
  "description": "string",
  "engineType": enum,
  "id": "string",
  "arn": "string",
  "latestRevision": {
    "created": "string",
    "description": "string",
    "revision": integer
  },
  "tags": {
  }
}
```

UpdateConfigurationOutput schema

```
{
  "created": "string",
  "warnings": [
    {
      "reason": enum,
      "attributeName": "string",
      "elementName": "string"
    }
  ],
  "name": "string",
  "id": "string",
  "arn": "string",
  "latestRevision": {
    "created": "string",
    "description": "string",
    "revision": integer
  }
}
```

Error schema

```
{
  "errorAttribute": "string",
  "message": "string"
}
```

Properties

AuthenticationStrategy

Optional. The authentication strategy used to secure the broker. The default is SIMPLE.

```
SIMPLE
LDAP
```

Configuration

Returns information about all configurations.

engineVersion

The broker engine version. Defaults to the latest available version for the specified broker engine type. For a list of supported engine versions, see the ActiveMQ version management and the RabbitMQ version management sections in the Amazon MQ Developer Guide.

Type: string
Required: True

created

Required. The date and time of the configuration revision.

Type: string
Required: True
Format: date-time

authenticationStrategy

Optional. The authentication strategy associated with the configuration. The default is SIMPLE.

Type: AuthenticationStrategy

Required: True

name

Required. The name of the configuration. This value can contain only alphanumeric characters, dashes, periods, underscores, and tildes (- . _ ~). This value must be 1-150 characters long.

    Type: string

    Required: True

description

Required. The description of the configuration.

    Type: string

    Required: True

engineType

Required. The type of broker engine. Currently, Amazon MQ supports ACTIVEMQ and RABBITMQ.

    Type: EngineType

    Required: True

id

Required. The unique ID that Amazon MQ generates for the configuration.

    Type: string

    Required: True

arn

Required. The ARN of the configuration.

    Type: string

    Required: True

latestRevision

Required. The latest revision of the configuration.

> **Type**: [ConfigurationRevision](#)
> **Required**: True

tags

The list of all tags associated with this configuration.

> **Type**: object
> **Required**: False

ConfigurationRevision

Returns information about the specified configuration revision.

created

Required. The date and time of the configuration revision.

> **Type**: string
> **Required**: True
> **Format**: date-time

description

The description of the configuration revision.

> **Type**: string
> **Required**: False

revision

Required. The revision number of the configuration.

> **Type**: integer
> **Required**: True

EngineType

The type of broker engine. Amazon MQ supports ActiveMQ and RabbitMQ.

```
ACTIVEMQ
RABBITMQ
```

Error

Returns information about an error.

errorAttribute

The attribute which caused the error.

Type: string
Required: False

message

The explanation of the error.

Type: string
Required: False

SanitizationWarning

Returns information about the configuration element or attribute that was sanitized in the configuration.

reason

The reason for which the configuration elements or attributes were sanitized.

Type: SanitizationWarningReason
Required: True

attributeName

The name of the configuration attribute that has been sanitized.

Type: string

Required: False

elementName

The name of the configuration element that has been sanitized.

Type: string

Required: False

SanitizationWarningReason

The reason for which the configuration elements or attributes were sanitized.

```
DISALLOWED_ELEMENT_REMOVED
DISALLOWED_ATTRIBUTE_REMOVED
INVALID_ATTRIBUTE_VALUE_REMOVED
```

UpdateConfigurationInput

Updates the specified configuration.

data

Amazon MQ for Active MQ: The base64-encoded XML configuration. Amazon MQ for RabbitMQ: the base64-encoded Cuttlefish configuration.

Type: string

Required: True

Format: byte

description

The description of the configuration.

Type: string

Required: False

UpdateConfigurationOutput

Returns information about the updated configuration.

created

Required. The date and time of the configuration.

> **Type**: string
> **Required**: True
> **Format**: date-time

warnings

The list of the first 20 warnings about the configuration elements or attributes that were sanitized.

> **Type**: Array of type [SanitizationWarning](#)
> **Required**: False

name

The name of the configuration. This value can contain only alphanumeric characters, dashes, periods, underscores, and tildes (- . _ ~). This value must be 1-150 characters long.

> **Type**: string
> **Required**: True

id

The unique ID that Amazon MQ generates for the configuration.

> **Type**: string
> **Required**: True

arn

The Amazon Resource Name (ARN) of the configuration.

> **Type**: string
> **Required**: True

latestRevision

The latest revision of the configuration.

> **Type**: ConfigurationRevision
> **Required**: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

DescribeConfiguration

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateConfiguration

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- AWS SDK for PHP V3
- AWS SDK for Python

- [AWS SDK for Ruby V3](#)

Configuration Revision

To keep track of the changes you make to your configuration, you can create configuration revisions. For more information, see [Configuration](#) in the Amazon MQ Developer Guide.

> ⚠️ **Important**
>
> Making changes to a configuration does not apply the changes to the broker immediately. To apply your changes, you must wait for the next maintenance window or reboot the broker.

URI

/v1/configurations/*configuration-id*/revisions/*configuration-revision*

HTTP methods

GET

Operation ID: `DescribeConfigurationRevision`

Returns the specified configuration revision for the specified configuration.

Path parameters

Name	Type	Required	Description
configuration-revision	String	True	The revision of the configuration.
configuration-id	String	True	The unique ID that Amazon MQ generates for the configuration.

Responses

Status code	Response model	Description
200	DescribeConfigurat ionRevisionOutput	HTTP Status Code 200: OK.
400	Error	HTTP Status Code 400: Bad request due to incorrect input. Correct your request and then retry it.
403	Error	HTTP Status Code 403: Access forbidden. Correct your credentials and then retry your request.
404	Error	HTTP Status Code 404: Resource not found due to incorrect input. Correct your request and then retry it.
500	Error	HTTP Status Code 500: Unexpected internal server error. Retrying your request might resolve the issue.

OPTIONS

Path parameters

Name	Type	Required	Description
configuration- revision	String	True	The revision of the configuration.
configuration- id	String	True	The unique ID that Amazon MQ

Name	Type	Required	Description
			generates for the configuration.

Responses

Status code	Response model	Description
200	None	Default response for CORS method

Schemas

Response bodies

DescribeConfigurationRevisionOutput schema

```
{
  "data": "string",
  "created": "string",
  "description": "string",
  "configurationId": "string"
}
```

Error schema

```
{
  "errorAttribute": "string",
  "message": "string"
}
```

Properties

DescribeConfigurationRevisionOutput

Returns the specified configuration revision for the specified configuration.

data

Amazon MQ for ActiveMQ: the base64-encoded XML configuration. Amazon MQ for RabbitMQ: base64-encoded Cuttlefish.

Type: string

Required: True

Format: byte

created

Required. The date and time of the configuration.

Type: string

Required: True

Format: date-time

description

The description of the configuration.

Type: string

Required: False

configurationId

Required. The unique ID that Amazon MQ generates for the configuration.

Type: string

Required: True

Error

Returns information about an error.

errorAttribute

The attribute which caused the error.

Type: string

Required: False

message

The explanation of the error.

Type: string

Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

DescribeConfigurationRevision

- [AWS Command Line Interface](#)

- [AWS SDK for .NET](#)

- [AWS SDK for C++](#)

- [AWS SDK for Go v2](#)

- [AWS SDK for Java V2](#)

- [AWS SDK for JavaScript V3](#)

- [AWS SDK for PHP V3](#)

- [AWS SDK for Python](#)

- [AWS SDK for Ruby V3](#)

Configuration Revisions

This is a collection of configuration revisions. To keep track of the changes you make to your configuration, you can create configuration revisions. For more information, see [Configuration](#) in the Amazon MQ Developer Guide.

> ⚠️ **Important**
>
> Making changes to a configuration does not apply the changes to the broker immediately. To apply your changes, you must wait for the next maintenance window or reboot the broker.

URI

/v1/configurations/*configuration-id*/revisions

HTTP methods

GET

Operation ID: ListConfigurationRevisions

Returns a list of all revisions for the specified configuration.

Path parameters

Name	Type	Required	Description
configuration-id	String	True	The unique ID that Amazon MQ generates for the configuration.

Query parameters

Name	Type	Required	Description
nextToken	String	False	The token that specifies the next page of results Amazon MQ should return. To request the first page, leave nextToken empty.

Name	Type	Required	Description
maxResults	String	False	The maximum number of brokers that Amazon MQ can return per page (20 by default). This value must be an integer from 5 to 100.

Responses

Status code	Response model	Description
200	ListConfigurationRevisionsOutput	HTTP Status Code 200: OK.
400	Error	HTTP Status Code 400: Bad request due to incorrect input. Correct your request and then retry it.
403	Error	HTTP Status Code 403: Access forbidden. Correct your credentials and then retry your request.
404	Error	HTTP Status Code 404: Resource not found due to incorrect input. Correct your request and then retry it.
500	Error	HTTP Status Code 500: Unexpected internal server error. Retrying your request might resolve the issue.

OPTIONS

Path parameters

Name	Type	Required	Description
configuration-id	String	True	The unique ID that Amazon MQ generates for the configuration.

Responses

Status code	Response model	Description
200	None	Default response for CORS method

Schemas

Response bodies

ListConfigurationRevisionsOutput schema

```
{
  "nextToken": "string",
  "maxResults": integer,
  "revisions": [
    {
      "created": "string",
      "description": "string",
      "revision": integer
    }
  ],
  "configurationId": "string"
}
```

Error schema

```
{
  "errorAttribute": "string",
  "message": "string"
}
```

Properties

ConfigurationRevision

Returns information about the specified configuration revision.

created

Required. The date and time of the configuration revision.

> **Type**: string
>
> **Required**: True
>
> **Format**: date-time

description

The description of the configuration revision.

> **Type**: string
>
> **Required**: False

revision

Required. The revision number of the configuration.

> **Type**: integer
>
> **Required**: True

Error

Returns information about an error.

errorAttribute

The attribute which caused the error.

> **Type**: string
> **Required**: False

message

The explanation of the error.

> **Type**: string
> **Required**: False

ListConfigurationRevisionsOutput

Returns a list of all revisions for the specified configuration.

nextToken

The token that specifies the next page of results Amazon MQ should return. To request the first page, leave nextToken empty.

> **Type**: string
> **Required**: False

maxResults

The maximum number of configuration revisions that can be returned per page (20 by default). This value must be an integer from 5 to 100.

> **Type**: integer
> **Required**: False

revisions

The list of all revisions for the specified configuration.

> **Type**: Array of type ConfigurationRevision

Required: False

configurationId

The unique ID that Amazon MQ generates for the configuration.

> **Type**: string
>
> **Required**: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

ListConfigurationRevisions

- [AWS Command Line Interface](#)

- [AWS SDK for .NET](#)

- [AWS SDK for C++](#)

- [AWS SDK for Go v2](#)

- [AWS SDK for Java V2](#)

- [AWS SDK for JavaScript V3](#)

- [AWS SDK for PHP V3](#)

- [AWS SDK for Python](#)

- [AWS SDK for Ruby V3](#)

Configurations

This is a collection of configurations. A configuration contains all of the settings for your broker. For more information, see [Configuration](#) and [Amazon MQ Broker Configuration Parameters](#) in the Amazon MQ Developer Guide.

You can create a configuration before creating any brokers. You can then apply the configuration to one or more brokers.

> **⚠ Important**
>
> Making changes to a configuration does not apply the changes to the broker immediately. To apply your changes, you must wait for the next maintenance window or reboot the broker.

URI

`/v1/configurations`

HTTP methods

GET

Operation ID: `ListConfigurations`

Returns a list of all configurations.

Query parameters

Name	Type	Required	Description
nextToken	String	False	The token that specifies the next page of results Amazon MQ should return. To request the first page, leave nextToken empty.
maxResults	String	False	The maximum number of brokers that Amazon MQ can return per page (20 by default). This value must be an integer from 5 to 100.

Responses

Status code	Response model	Description
200	<u>ListConfigurations Output</u>	HTTP Status Code 200: OK.
400	<u>Error</u>	HTTP Status Code 400: Bad request due to incorrect input. Correct your request and then retry it.
403	<u>Error</u>	HTTP Status Code 403: Access forbidden. Correct your credentials and then retry your request.
500	<u>Error</u>	HTTP Status Code 500: Unexpected internal server error. Retrying your request might resolve the issue.

POST

Operation ID: CreateConfiguration

Creates a new configuration for the specified configuration name. Amazon MQ uses the default configuration (the engine type and version).

Responses

Status code	Response model	Description
200	<u>CreateConfiguratio nOutput</u>	HTTP Status Code 200: OK.
400	<u>Error</u>	HTTP Status Code 400: Bad request due to incorrect

Status code	Response model	Description
		input. Correct your request and then retry it.
403	[Error](#)	HTTP Status Code 403: Access forbidden. Correct your credentials and then retry your request.
409	[Error](#)	HTTP Status Code 409: Conflict. This broker name already exists. Retry your request with another name.
500	[Error](#)	HTTP Status Code 500: Unexpected internal server error. Retrying your request might resolve the issue.

OPTIONS

Responses

Status code	Response model	Description
200	None	Default response for CORS method

Schemas

Request bodies

POST schema

```
{
  "engineVersion": "string",
  "authenticationStrategy": enum,
```

```
    "name": "string",
    "engineType": enum,
    "tags": {
    }
  }
}
```

Response bodies

ListConfigurationsOutput schema

```
{
  "nextToken": "string",
  "maxResults": integer,
  "configurations": [
    {
      "engineVersion": "string",
      "created": "string",
      "authenticationStrategy": enum,
      "name": "string",
      "description": "string",
      "engineType": enum,
      "id": "string",
      "arn": "string",
      "latestRevision": {
        "created": "string",
        "description": "string",
        "revision": integer
      },
      "tags": {
      }
    }
  ]
}
```

CreateConfigurationOutput schema

```
{
  "created": "string",
  "authenticationStrategy": enum,
  "name": "string",
  "id": "string",
  "arn": "string",
```

```
    "latestRevision": {
      "created": "string",
      "description": "string",
      "revision": integer
    }
  }
}
```

Error schema

```
{
  "errorAttribute": "string",
  "message": "string"
}
```

Properties

AuthenticationStrategy

Optional. The authentication strategy used to secure the broker. The default is SIMPLE.

```
SIMPLE
LDAP
```

Configuration

Returns information about all configurations.

engineVersion

The broker engine version. Defaults to the latest available version for the specified broker engine type. For a list of supported engine versions, see the [ActiveMQ version management](#) and the [RabbitMQ version management](#) sections in the Amazon MQ Developer Guide.

Type: string
Required: True

created

Required. The date and time of the configuration revision.

Type: string

Required: True

Format: date-time

authenticationStrategy

Optional. The authentication strategy associated with the configuration. The default is SIMPLE.

Type: [AuthenticationStrategy](#)

Required: True

name

Required. The name of the configuration. This value can contain only alphanumeric characters, dashes, periods, underscores, and tildes (- . _ ~). This value must be 1-150 characters long.

Type: string

Required: True

description

Required. The description of the configuration.

Type: string

Required: True

engineType

Required. The type of broker engine. Currently, Amazon MQ supports ACTIVEMQ and RABBITMQ.

Type: [EngineType](#)

Required: True

id

Required. The unique ID that Amazon MQ generates for the configuration.

Type: string

Required: True

arn

Required. The ARN of the configuration.

> **Type**: string
> **Required**: True

latestRevision

Required. The latest revision of the configuration.

> **Type**: [ConfigurationRevision](#)
> **Required**: True

tags

The list of all tags associated with this configuration.

> **Type**: object
> **Required**: False

ConfigurationRevision

Returns information about the specified configuration revision.

created

Required. The date and time of the configuration revision.

> **Type**: string
> **Required**: True
> **Format**: date-time

description

The description of the configuration revision.

Type: string

Required: False

revision

Required. The revision number of the configuration.

Type: integer

Required: True

CreateConfigurationInput

Creates a new configuration for the specified configuration name. Amazon MQ uses the default configuration (the engine type and version).

engineVersion

The broker engine version. Defaults to the latest available version for the specified broker engine type. For more information, see the [ActiveMQ version management](#) and the [RabbitMQ version management](#) sections in the Amazon MQ Developer Guide.

Type: string

Required: False

authenticationStrategy

Optional. The authentication strategy associated with the configuration. The default is SIMPLE.

Type: [AuthenticationStrategy](#)

Required: False

name

Required. The name of the configuration. This value can contain only alphanumeric characters, dashes, periods, underscores, and tildes (- . _ ~). This value must be 1-150 characters long.

Type: string

Required: True

engineType

Required. The type of broker engine. Currently, Amazon MQ supports ACTIVEMQ and RABBITMQ.

> **Type**: [EngineType](#)
> **Required**: True

tags

Create tags when creating the configuration.

> **Type**: object
> **Required**: False

CreateConfigurationOutput

Returns information about the created configuration.

created

Required. The date and time of the configuration.

> **Type**: string
> **Required**: True
> **Format**: date-time

authenticationStrategy

Optional. The authentication strategy associated with the configuration. The default is SIMPLE.

> **Type**: [AuthenticationStrategy](#)
> **Required**: True

name

Required. The name of the configuration. This value can contain only alphanumeric characters, dashes, periods, underscores, and tildes (- . _ ~). This value must be 1-150 characters long.

> **Type**: string
> **Required**: True

id

Required. The unique ID that Amazon MQ generates for the configuration.

   Type: string
   Required: True

arn

Required. The Amazon Resource Name (ARN) of the configuration.

   Type: string
   Required: True

latestRevision

The latest revision of the configuration.

   Type: ConfigurationRevision
   Required: False

EngineType

The type of broker engine. Amazon MQ supports ActiveMQ and RabbitMQ.

   ACTIVEMQ
   RABBITMQ

Error

Returns information about an error.

errorAttribute

The attribute which caused the error.

   Type: string
   Required: False

message

The explanation of the error.

> **Type**: string
> **Required**: False

ListConfigurationsOutput

Returns a list of all configurations.

nextToken

The token that specifies the next page of results Amazon MQ should return. To request the first page, leave nextToken empty.

> **Type**: string
> **Required**: False

maxResults

The maximum number of configurations that Amazon MQ can return per page (20 by default). This value must be an integer from 5 to 100.

> **Type**: integer
> **Required**: False

configurations

The list of all revisions for the specified configuration.

> **Type**: Array of type Configuration
> **Required**: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

ListConfigurations

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

CreateConfiguration

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

Tag

A tag is a key-value pair associated with a resource. You can use these metadata tags to identify the purpose of a broker or configuration. For more information see [Tagging resources](#) in the Amazon MQ Developer Guide.

URI

`/v1/tags/`*`resource-arn`*

HTTP methods

GET

Operation ID: ListTags

Lists tags for a resource.

Path parameters

Name	Type	Required	Description
resource-arn	String	True	The Amazon Resource Name (ARN) of the resource tag.

Responses

Status code	Response model	Description
200	Tags	HTTP Status Code 200: OK.
400	Error	HTTP Status Code 400: Bad request due to incorrect input. Correct your request and then retry it.
403	Error	HTTP Status Code 403: Access forbidden. Correct your credentials and then retry your request.
404	Error	HTTP Status Code 404: Resource not found due to incorrect input. Correct your request and then retry it.
500	Error	HTTP Status Code 500: Unexpected internal server

Status code	Response model	Description
		error. Retrying your request might resolve the issue.

POST

Operation ID: `CreateTags`

Add a tag to a resource.

Path parameters

Name	Type	Required	Description
resource-arn	String	True	The Amazon Resource Name (ARN) of the resource tag.

Responses

Status code	Response model	Description
204	None	HTTP Status Code 204: Successful response.
400	[Error]	HTTP Status Code 400: Bad request due to incorrect input. Correct your request and then retry it.
403	[Error]	HTTP Status Code 403: Access forbidden. Correct your credentials and then retry your request.
404	[Error]	HTTP Status Code 404: Resource not found due to

Status code	Response model	Description
		incorrect input. Correct your request and then retry it.
500	Error	HTTP Status Code 500: Unexpected internal server error. Retrying your request might resolve the issue.

DELETE

Operation ID: `DeleteTags`

Removes a tag from a resource.

Path parameters

Name	Type	Required	Description
resource-arn	String	True	The Amazon Resource Name (ARN) of the resource tag.

Query parameters

Name	Type	Required	Description
tagKeys	String	True	An array of tag keys to delete

Responses

Status code	Response model	Description
204	None	HTTP Status Code 204: Successful response.

Status code	Response model	Description
400	[Error](#)	HTTP Status Code 400: Bad request due to incorrect input. Correct your request and then retry it.
403	[Error](#)	HTTP Status Code 403: Access forbidden. Correct your credentials and then retry your request.
404	[Error](#)	HTTP Status Code 404: Resource not found due to incorrect input. Correct your request and then retry it.
500	[Error](#)	HTTP Status Code 500: Unexpected internal server error. Retrying your request might resolve the issue.

OPTIONS

Path parameters

Name	Type	Required	Description
resource-arn	String	True	The Amazon Resource Name (ARN) of the resource tag.

Responses

Status code	Response model	Description
200	None	Default response for CORS method

Schemas

Request bodies

POST schema

```
{
  "tags": {
  }
}
```

Response bodies

Tags schema

```
{
  "tags": {
  }
}
```

Error schema

```
{
  "errorAttribute": "string",
  "message": "string"
}
```

Properties

Error

Returns information about an error.

errorAttribute

The attribute which caused the error.

    Type: string

    Required: False

message

The explanation of the error.

    Type: string

    Required: False

Tags

A map of the key-value pairs for the resource tag.

tags

The key-value pair for the resource tag.

    Type: object

    Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

ListTags

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)

- AWS SDK for JavaScript V3

- AWS SDK for PHP V3

- AWS SDK for Python

- AWS SDK for Ruby V3

CreateTags

- AWS Command Line Interface

- AWS SDK for .NET

- AWS SDK for C++

- AWS SDK for Go v2

- AWS SDK for Java V2

- AWS SDK for JavaScript V3

- AWS SDK for PHP V3

- AWS SDK for Python

- AWS SDK for Ruby V3

DeleteTags

- AWS Command Line Interface

- AWS SDK for .NET

- AWS SDK for C++

- AWS SDK for Go v2

- AWS SDK for Java V2

- AWS SDK for JavaScript V3

- AWS SDK for PHP V3

- AWS SDK for Python

- AWS SDK for Ruby V3

User

> **ⓘ Note**
>
> Does not apply to RabbitMQ brokers.

An ActiveMQ user is a person or an application that can access the queues and topics of an ActiveMQ broker. For more information, see [User](#) in the Amazon MQ Developer Guide.

A user can belong to a group. You can configure which users belong to which groups and which groups have permission to send to, receive from, and administer specific queues and topics.

> **⚠ Important**
>
> Making changes to a user does not apply the changes to the user immediately. To apply your changes, you must wait for the next maintenance window or reboot the broker.

URI

`/v1/brokers/`*`broker-id`*`/users/`*`username`*

HTTP methods

GET

Operation ID: `DescribeUser`

Returns information about an ActiveMQ user.

Path parameters

Name	Type	Required	Description
username	String	True	The username of the ActiveMQ user. This value can contain only alphanumeric characters, dashes,

Name	Type	Required	Description
			periods, underscor es, and tildes (- . _ ~). This value must be 2-100 characters long.
broker-id	String	True	The unique ID that Amazon MQ generates for the broker.

Responses

Status code	Response model	Description
200	DescribeUserOutput	HTTP Status Code 200: OK.
400	Error	HTTP Status Code 400: Bad request due to incorrect input. Correct your request and then retry it.
403	Error	HTTP Status Code 403: Access forbidden. Correct your credentials and then retry your request.
404	Error	HTTP Status Code 404: Resource not found due to incorrect input. Correct your request and then retry it.
500	Error	HTTP Status Code 500: Unexpected internal server error. Retrying your request might resolve the issue.

POST

Operation ID: `CreateUser`

Creates an ActiveMQ user.

> ⚠️ **Important**
>
> Do not add personally identifiable information (PII) or other confidential or sensitive information in broker usernames. Broker usernames are accessible to other AWS services, including CloudWatch Logs. Broker usernames are not intended to be used for private or sensitive data.

Path parameters

Name	Type	Required	Description
username	String	True	The username of the ActiveMQ user. This value can contain only alphanumeric characters, dashes, periods, underscores, and tildes (- . _ ~). This value must be 2-100 characters long.
broker-id	String	True	The unique ID that Amazon MQ generates for the broker.

Responses

Status code	Response model	Description
200	None	HTTP Status Code 200: OK.

Status code	Response model	Description
400	Error	HTTP Status Code 400: Bad request due to incorrect input. Correct your request and then retry it.
403	Error	HTTP Status Code 403: Access forbidden. Correct your credentials and then retry your request.
404	Error	HTTP Status Code 404: Resource not found due to incorrect input. Correct your request and then retry it.
409	Error	HTTP Status Code 409: Conflict. This broker name already exists. Retry your request with another name.
500	Error	HTTP Status Code 500: Unexpected internal server error. Retrying your request might resolve the issue.

PUT

Operation ID: `UpdateUser`

Updates the information for an ActiveMQ user.

Path parameters

Name	Type	Required	Description
username	String	True	The username of the ActiveMQ user. This

Name	Type	Required	Description
			value can contain only alphanumeric characters, dashes, periods, underscor es, and tildes (- . _ ~). This value must be 2-100 characters long.
broker-id	String	True	The unique ID that Amazon MQ generates for the broker.

Responses

Status code	Response model	Description
200	None	HTTP Status Code 200: OK.
400	Error	HTTP Status Code 400: Bad request due to incorrect input. Correct your request and then retry it.
403	Error	HTTP Status Code 403: Access forbidden. Correct your credentials and then retry your request.
404	Error	HTTP Status Code 404: Resource not found due to incorrect input. Correct your request and then retry it.

Status code	Response model	Description
409	Error	HTTP Status Code 409: Conflict. This broker name already exists. Retry your request with another name.
500	Error	HTTP Status Code 500: Unexpected internal server error. Retrying your request might resolve the issue.

DELETE

Operation ID: `DeleteUser`

Deletes an ActiveMQ user.

Path parameters

Name	Type	Required	Description
username	String	True	The username of the ActiveMQ user. This value can contain only alphanumeric characters, dashes, periods, underscores, and tildes (- . _ ~). This value must be 2-100 characters long.
broker-id	String	True	The unique ID that Amazon MQ generates for the broker.

Responses

Status code	Response model	Description
200	None	HTTP Status Code 200: OK.
400	[Error](#)	HTTP Status Code 400: Bad request due to incorrect input. Correct your request and then retry it.
403	[Error](#)	HTTP Status Code 403: Access forbidden. Correct your credentials and then retry your request.
404	[Error](#)	HTTP Status Code 404: Resource not found due to incorrect input. Correct your request and then retry it.
500	[Error](#)	HTTP Status Code 500: Unexpected internal server error. Retrying your request might resolve the issue.

OPTIONS

Path parameters

Name	Type	Required	Description
username	String	True	The username of the ActiveMQ user. This value can contain only alphanumeric characters, dashes, periods, underscores, and tildes (- . _

Name	Type	Required	Description
			~). This value must be 2-100 characters long.
broker-id	String	True	The unique ID that Amazon MQ generates for the broker.

Responses

Status code	Response model	Description
200	None	Default response for CORS method

Schemas

Request bodies

POST schema

```
{
  "password": "string",
  "replicationUser": boolean,
  "groups": [
    "string"
  ],
  "consoleAccess": boolean
}
```

PUT schema

```
{
  "password": "string",
  "replicationUser": boolean,
```

```
    "groups": [
      "string"
    ],
    "consoleAccess": boolean
}
```

Response bodies

DescribeUserOutput schema

```
{
  "brokerId": "string",
  "replicationUser": boolean,
  "pending": {
    "pendingChange": enum,
    "groups": [
      "string"
    ],
    "consoleAccess": boolean
  },
  "groups": [
    "string"
  ],
  "consoleAccess": boolean,
  "username": "string"
}
```

Error schema

```
{
  "errorAttribute": "string",
  "message": "string"
}
```

Properties

ChangeType

The type of change pending for the ActiveMQ user.

    CREATE

```
UPDATE
DELETE
```

CreateUserInput

Creates a new ActiveMQ user.

password

Required. The password of the user. This value must be at least 12 characters long, must contain at least 4 unique characters, and must not contain commas, colons, or equal signs (,:=).

> **Type**: string
> **Required**: True
> **Format**: password

replicationUser

Defines if this user is intended for CRDR replication purposes.

> **Type**: boolean
> **Required**: False

groups

The list of groups (20 maximum) to which the ActiveMQ user belongs. This value can contain only alphanumeric characters, dashes, periods, underscores, and tildes (- . _ ~). This value must be 2-100 characters long.

> **Type**: Array of type string
> **Required**: False

consoleAccess

Enables access to the ActiveMQ Web Console for the ActiveMQ user.

> **Type**: boolean
> **Required**: False

DescribeUserOutput

Returns information about an ActiveMQ user.

brokerId

Required. The unique ID that Amazon MQ generates for the broker.

> **Type**: string
> **Required**: True

replicationUser

Describes whether the user is intended for data replication replication

> **Type**: boolean
> **Required**: False

pending

The status of the changes pending for the ActiveMQ user.

> **Type**: UserPendingChanges
> **Required**: False

groups

The list of groups (20 maximum) to which the ActiveMQ user belongs. This value can contain only alphanumeric characters, dashes, periods, underscores, and tildes (- . _ ~). This value must be 2-100 characters long.

> **Type**: Array of type string
> **Required**: False

consoleAccess

Enables access to the the ActiveMQ Web Console for the ActiveMQ user.

> **Type**: boolean
> **Required**: False

username

Required. The username of the ActiveMQ user. This value can contain only alphanumeric characters, dashes, periods, underscores, and tildes (- . _ ~). This value must be 2-100 characters long.

> **Type**: string
> **Required**: True

Error

Returns information about an error.

errorAttribute

The attribute which caused the error.

> **Type**: string
> **Required**: False

message

The explanation of the error.

> **Type**: string
> **Required**: False

UpdateUserInput

Updates the information for an ActiveMQ user.

password

The password of the user. This value must be at least 12 characters long, must contain at least 4 unique characters, and must not contain commas, colons, or equal signs (,:=).

> **Type**: string
> **Required**: False
> **Format**: password

replicationUser

Defines whether the user is intended for data replication replication.

> **Type**: boolean
> **Required**: False

groups

The list of groups (20 maximum) to which the ActiveMQ user belongs. This value can contain only alphanumeric characters, dashes, periods, underscores, and tildes (- . _ ~). This value must be 2-100 characters long.

> **Type**: Array of type string
> **Required**: False

consoleAccess

Enables access to the the ActiveMQ Web Console for the ActiveMQ user.

> **Type**: boolean
> **Required**: False

UserPendingChanges

Returns information about the status of the changes pending for the ActiveMQ user.

pendingChange

Required. The type of change pending for the ActiveMQ user.

> **Type**: ChangeType
> **Required**: True

groups

The list of groups (20 maximum) to which the ActiveMQ user belongs. This value can contain only alphanumeric characters, dashes, periods, underscores, and tildes (- . _ ~). This value must be 2-100 characters long.

Type: Array of type string

Required: False

consoleAccess

Enables access to the the ActiveMQ Web Console for the ActiveMQ user.

Type: boolean

Required: False

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

DescribeUser

- [AWS Command Line Interface](#)

- [AWS SDK for .NET](#)

- [AWS SDK for C++](#)

- [AWS SDK for Go v2](#)

- [AWS SDK for Java V2](#)

- [AWS SDK for JavaScript V3](#)

- [AWS SDK for PHP V3](#)

- [AWS SDK for Python](#)

- [AWS SDK for Ruby V3](#)

CreateUser

- [AWS Command Line Interface](#)

- [AWS SDK for .NET](#)

- [AWS SDK for C++](#)

- [AWS SDK for Go v2](#)

- AWS SDK for Java V2

- AWS SDK for JavaScript V3

- AWS SDK for PHP V3

- AWS SDK for Python

- AWS SDK for Ruby V3

UpdateUser

- AWS Command Line Interface

- AWS SDK for .NET

- AWS SDK for C++

- AWS SDK for Go v2

- AWS SDK for Java V2

- AWS SDK for JavaScript V3

- AWS SDK for PHP V3

- AWS SDK for Python

- AWS SDK for Ruby V3

DeleteUser

- AWS Command Line Interface

- AWS SDK for .NET

- AWS SDK for C++

- AWS SDK for Go v2

- AWS SDK for Java V2

- AWS SDK for JavaScript V3

- AWS SDK for PHP V3

- AWS SDK for Python

- AWS SDK for Ruby V3

Users

> **ⓘ Note**
>
> Does not apply to RabbitMQ brokers.

This is a collection of ActiveMQ users for the specified broker. An ActiveMQ user is a person or an application that can access the queues and topics of an ActiveMQ broker. For more information, see User in the Amazon MQ Developer Guide.

You can configure to have specific permissions. For example, you can allow some users to access the ActiveMQ Web Console.

A user can belong to a group. You can configure which users belong to which groups and which groups have permission to send to, receive from, and administer specific queues and topics.

> **⚠ Important**
>
> Making changes to a user does not apply the changes to the user immediately. To apply your changes, you must wait for the next maintenance window or reboot the broker.

URI

/v1/brokers/*broker-id*/users

HTTP methods

GET

Operation ID: `ListUsers`

Returns a list of all ActiveMQ users.

Path parameters

Name	Type	Required	Description
broker-id	String	True	The unique ID that Amazon MQ

Name	Type	Required	Description
			generates for the broker.

Query parameters

Name	Type	Required	Description
nextToken	String	False	The token that specifies the next page of results Amazon MQ should return. To request the first page, leave nextToken empty.
maxResults	String	False	The maximum number of brokers that Amazon MQ can return per page (20 by default). This value must be an integer from 5 to 100.

Responses

Status code	Response model	Description
200	ListUsersOutput	HTTP Status Code 200: OK.
400	Error	HTTP Status Code 400: Bad request due to incorrect input. Correct your request and then retry it.
403	Error	HTTP Status Code 403: Access forbidden. Correct your

Status code	Response model	Description
		credentials and then retry your request.
404	[Error](#)	HTTP Status Code 404: Resource not found due to incorrect input. Correct your request and then retry it.
500	[Error](#)	HTTP Status Code 500: Unexpected internal server error. Retrying your request might resolve the issue.

OPTIONS

Path parameters

Name	Type	Required	Description
broker-id	String	True	The unique ID that Amazon MQ generates for the broker.

Responses

Status code	Response model	Description
200	None	Default response for CORS method

Schemas

Response bodies

ListUsersOutput schema

```
{
  "brokerId": "string",
  "nextToken": "string",
  "maxResults": integer,
  "users": [
    {
      "pendingChange": enum,
      "username": "string"
    }
  ]
}
```

Error schema

```
{
  "errorAttribute": "string",
  "message": "string"
}
```

Properties

ChangeType

The type of change pending for the ActiveMQ user.

```
CREATE
UPDATE
DELETE
```

Error

Returns information about an error.

errorAttribute

The attribute which caused the error.

Type: string
Required: False

message

The explanation of the error.

> **Type**: string
> **Required**: False

ListUsersOutput

Returns a list of all ActiveMQ users.

brokerId

Required. The unique ID that Amazon MQ generates for the broker.

> **Type**: string
> **Required**: True

nextToken

The token that specifies the next page of results Amazon MQ should return. To request the first page, leave nextToken empty.

> **Type**: string
> **Required**: False

maxResults

Required. The maximum number of ActiveMQ users that can be returned per page (20 by default). This value must be an integer from 5 to 100.

> **Type**: integer
> **Required**: True
> **Minimum**: 5
> **Maximum**: 100

users

Required. The list of all ActiveMQ usernames for the specified broker. Does not apply to RabbitMQ brokers.

Type: Array of type UserSummary

Required: True

UserSummary

Returns a list of all broker users. Does not apply to RabbitMQ brokers.

pendingChange

The type of change pending for the broker user.

Type: ChangeType

Required: False

username

Required. The username of the broker user. This value can contain only alphanumeric characters, dashes, periods, underscores, and tildes (- . _ ~). This value must be 2-100 characters long.

Type: string

Required: True

See also

For more information about using this API in one of the language-specific AWS SDKs and references, see the following:

ListUsers

- AWS Command Line Interface
- [AWS SDK for .NET](AWS SDK for .NET)
- [AWS SDK for C++](AWS SDK for C++)
- [AWS SDK for Go v2](AWS SDK for Go v2)
- AWS SDK for Java V2
- AWS SDK for JavaScript V3
- [AWS SDK for PHP V3](AWS SDK for PHP V3)
- AWS SDK for Python

- [AWS SDK for Ruby V3](#)

Operations

The Amazon MQ REST API includes the following operations.

- CreateBroker

  Creates a broker. Note: This API is asynchronous.

  To create a broker, you must either use the `AmazonMQFullAccess` IAM policy or include the following EC2 permissions in your IAM policy.

  - `ec2:CreateNetworkInterface`

    This permission is required to allow Amazon MQ to create an elastic network interface (ENI) on behalf of your account.

  - `ec2:CreateNetworkInterfacePermission`

    This permission is required to attach the ENI to the broker instance.

  - `ec2:DeleteNetworkInterface`

  - `ec2:DeleteNetworkInterfacePermission`

  - `ec2:DetachNetworkInterface`

  - `ec2:DescribeInternetGateways`

  - `ec2:DescribeNetworkInterfaces`

  - `ec2:DescribeNetworkInterfacePermissions`

  - `ec2:DescribeRouteTables`

  - `ec2:DescribeSecurityGroups`

  - `ec2:DescribeSubnets`

  - `ec2:DescribeVpcs`

  For more information, see Create an IAM User and Get Your AWS Credentials and Never Modify or Delete the Amazon MQ Elastic Network Interface in the *Amazon MQ Developer Guide*.

- CreateConfiguration

  Creates a new configuration for the specified configuration name. Amazon MQ uses the default configuration (the engine type and version).

- CreateTags

Add a tag to a resource.

- CreateUser

  Creates an ActiveMQ user.

> ⚠️ **Important**
>
> Do not add personally identifiable information (PII) or other confidential or sensitive information in broker usernames. Broker usernames are accessible to other AWS services, including CloudWatch Logs. Broker usernames are not intended to be used for private or sensitive data.

- DeleteBroker

  Deletes a broker. Note: This API is asynchronous.

- DeleteTags

  Removes a tag from a resource.

- DeleteUser

  Deletes an ActiveMQ user.

- DescribeBroker

  Returns information about the specified broker.

- DescribeBrokerEngineTypes

  Describe available engine types and versions.

- DescribeBrokerInstanceOptions

  Describe available broker instance options.

- DescribeConfiguration

  Returns information about the specified configuration.

- DescribeConfigurationRevision

  Returns the specified configuration revision for the specified configuration.

- DescribeUser

Returns information about an ActiveMQ user.

- ListBrokers

  Returns a list of all brokers.

- ListConfigurationRevisions

  Returns a list of all revisions for the specified configuration.

- ListConfigurations

  Returns a list of all configurations.

- ListTags

  Lists tags for a resource.

- ListUsers

  Returns a list of all ActiveMQ users.

- Promote

  Promotes a data replication replica broker to a primary.

- RebootBroker

  Reboots a broker. Note: This API is asynchronous.

- UpdateBroker

  Adds a pending configuration change to a broker.

- UpdateConfiguration

  Updates the specified configuration.

- UpdateUser

  Updates the information for an ActiveMQ user.