Whizlabs Offer | Flat 15% OFF SITEWIDE | Use Coupon - WHIZOFFER15

> My Courses > AWS Certified Solutions Architect Associate > STS, SNS, SQS - Quiz > **Report**

Search Courses

STS, SNS, SQS - Quiz

Completed on 09-January-2020

Attempt

04

Marks Obtained

2 / 10

Your score

20%

Time Taken

00 H 00 M 35 S

Result

Failed

Domains wise Quiz Performance Report

No	1
Domain	Design Resilient Architectures
Total Question	2
Correct	0
Incorrect	2
Unattempted	0
Marked for review	0

No	2
Domain	Other
Total Question	8
Correct	2
Incorrect	6
Unattempted	0
Marked for review	0
Total	Total
All Domain	All Domain
Total Question	10
Correct	2
Incorrect	8
Unattempted	0
Marked for review	0

Review the Answers

Sorting by All

Question 1 Incorrect

Domain :Design Resilient Architectures

Company ABC has an AWS setup and planning to use Amazon SQS for queuing messages. The design is such that two applications will receive the same message in the queue and process it. Once applications would have read the message, it should be deleted. However, when the 2nd application is making ReceiveMessage API call, the message is not getting returned. Which of the following could be reasons? (Choose 2 options)

✓ A. **Application 2 is making a call before Visibility Timeout elapsed which was set by application 1 ReceiveMessage call.** ✅

✓ B. **Amazon SQS deletes the message once it has been responded via ReceiveMessage call from Application 1.** ❌

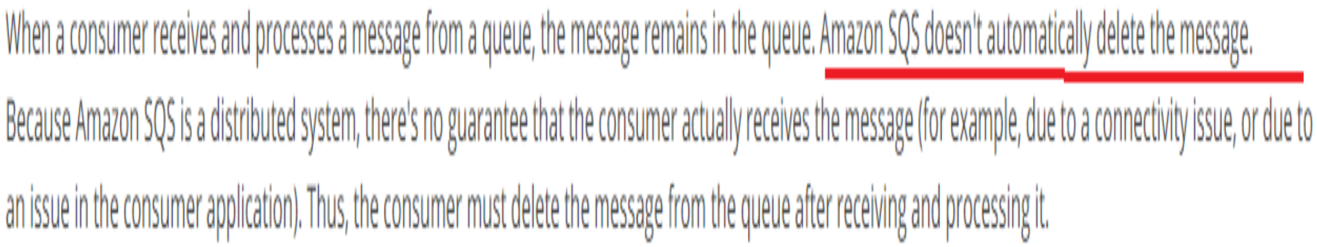
C. **Application 1 had deleted the message after it has processed before Visibility Timeout elapsed.** ✅

D. **Application 2 does not have access on the message it is trying to receive.**

Explanation:

Answer: A, C

Option A is correct.

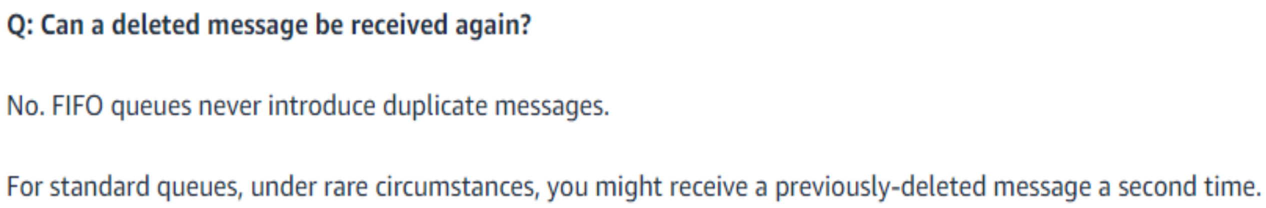


When a consumer receives and processes a message from a queue, the message remains in the queue. Amazon SQS doesn't automatically delete the message. Because Amazon SQS is a distributed system, there's no guarantee that the consumer actually receives the message (for example, due to a connectivity issue, or due to an issue in the consumer application). Thus, the consumer must delete the message from the queue after receiving and processing it.

Immediately after a message is received, it remains in the queue. To prevent other consumers from processing the message again, Amazon SQS sets a visibility timeout, a period of time during which Amazon SQS prevents other consumers from receiving and processing the message. The default visibility timeout for a message is 30 seconds. The minimum is 0 seconds. The maximum is 12 hours.

https://docs.aws.amazon.com/AWSSimpleQueueService/latest/SQSDeveloperGuide/sqs-visibility-timeout.html

Option B is not correct.



Q: Can a deleted message be received again?

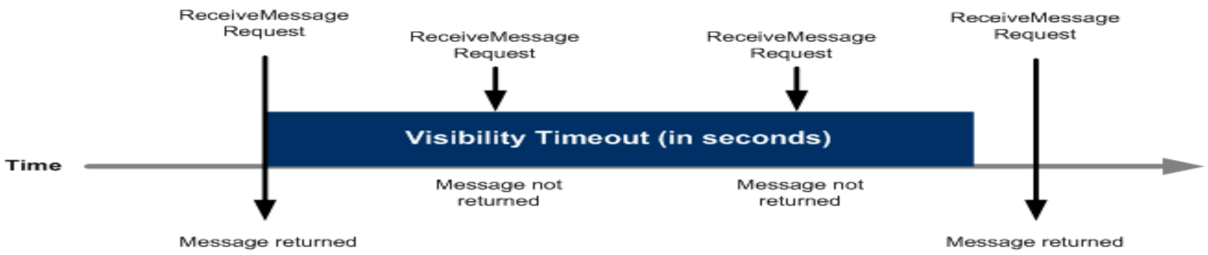
No. FIFO queues never introduce duplicate messages.

For standard queues, under rare circumstances, you might receive a previously-deleted message a second time.

https://docs.aws.amazon.com/AWSSimpleQueueService/latest/SQSDeveloperGuide/sqs-visibility-timeout.html

Option C is correct.

Once the message is deleted from Amazon SQS Queue, it will not be available anymore.



Option D is not correct.

Permission exists on the Queue level, not on the message level.

For more information on SQS actions, refer documentation here.

https://docs.aws.amazon.com/IAM/latest/UserGuide/list_amazonsqs.html

Ask our Experts

Rate this Question? ☺ ☹

Question 2

Incorrect

Domain :Design Resilient Architectures

Your organization is using Amazon SQS as an enterprise message queuing platform. 100s of applications reading the queues every few seconds to process the messages and delete them as soon as they are being written into the queues. Looking at the number of requests being sent to Amazon SQS APIs, your management is concerned on the pricing that will be incurred. As an architect, how would you reduce pricing without compromising on time in this scenario? Please select 2 correct answers.

✓ A. **Once successfully written, Amazon SQS messages are only available after 1 minute. Ask applications to increase the delay between calls to 1 minute. This reduces the number of API calls made** ✗

✓ B. **Use Amazon SQS Long Polling.** ✓

C. **Send DeleteMessage requests in batch.** ✓

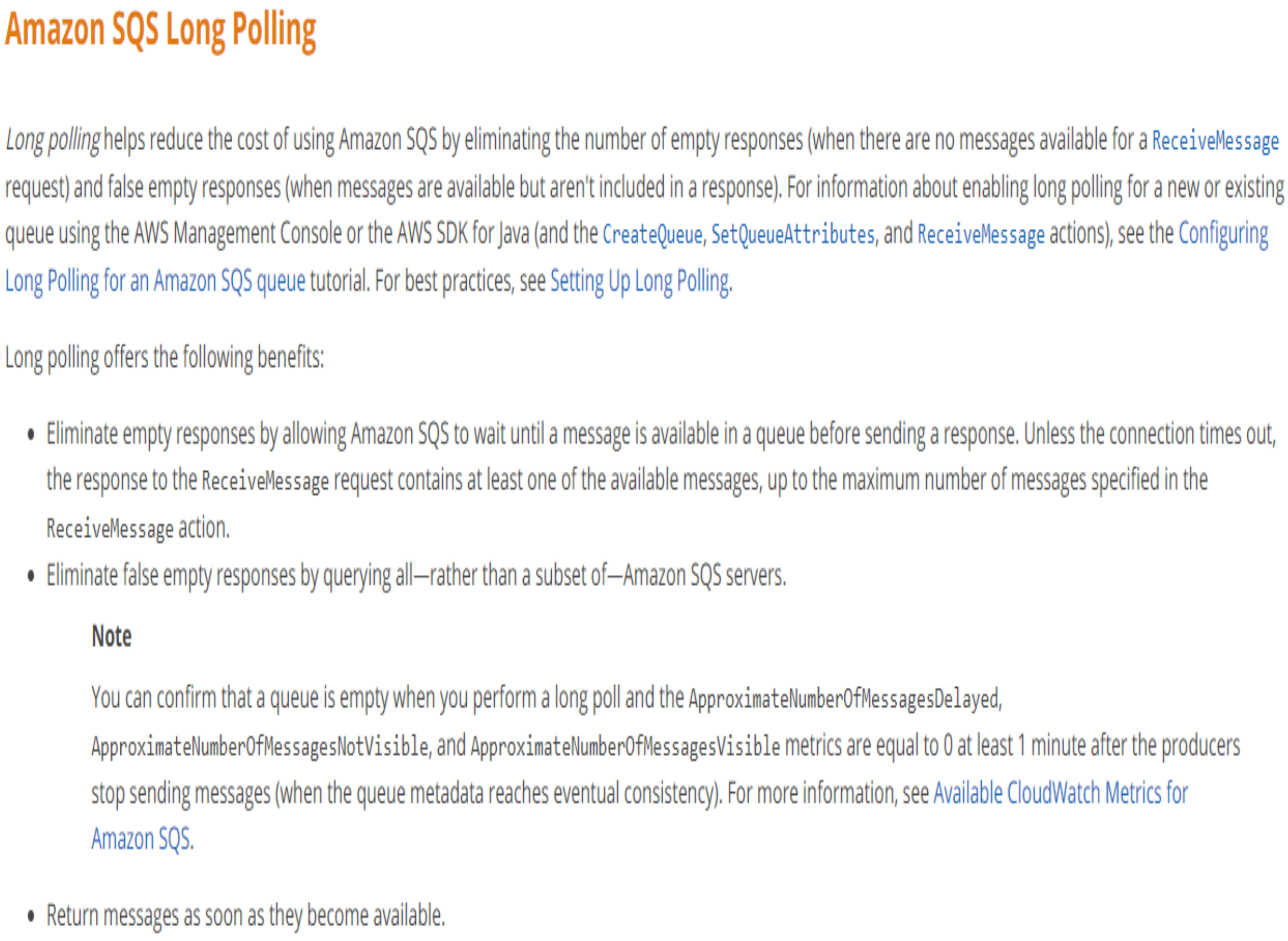
D. **Use Amazon SQS Short Polling.**

Explanation:

Answer: B, C

Option A is incorrect. There is no such limitation on AWS SQS queues.

Option B is correct.



Amazon SQS Long Polling

Long polling helps reduce the cost of using Amazon SQS by eliminating the number of empty responses (when there are no messages available for a `ReceiveMessage` request) and false empty responses (when messages are available but aren't included in a response). For information about enabling long polling for a new or existing queue using the AWS Management Console or the AWS SDK for Java (and the `CreateQueue`, `SetQueueAttributes`, and `ReceiveMessage` actions), see the Configuring Long Polling for an Amazon SQS queue tutorial. For best practices, see Setting Up Long Polling.

Long polling offers the following benefits:

- Eliminate empty responses by allowing Amazon SQS to wait until a message is available in a queue before sending a response. Unless the connection times out, the response to the `ReceiveMessage` request contains at least one of the available messages, up to the maximum number of messages specified in the `ReceiveMessage` action.
- Eliminate false empty responses by querying all—rather than a subset of—Amazon SQS servers.

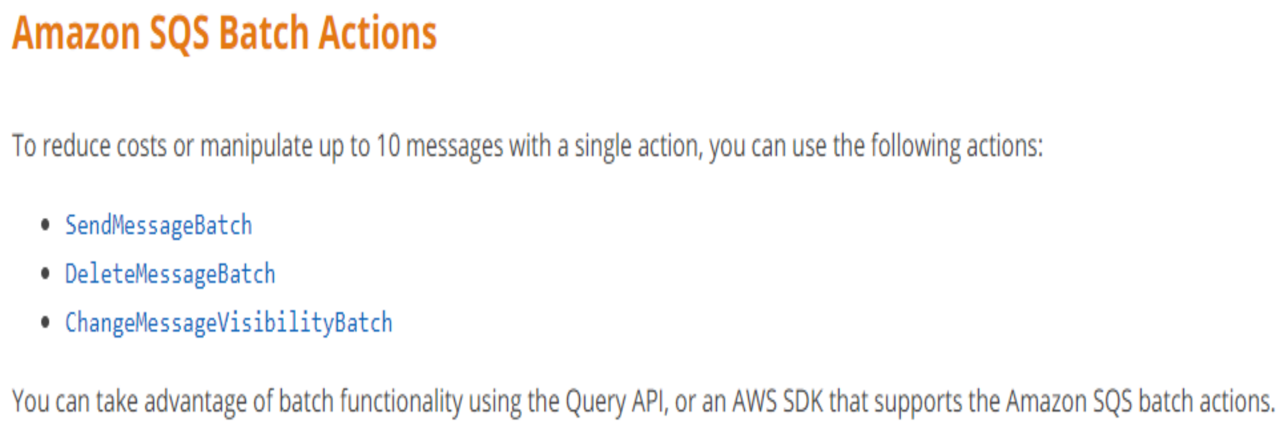
  Note

  You can confirm that a queue is empty when you perform a long poll and the `ApproximateNumberOfMessagesDelayed`, `ApproximateNumberOfMessagesNotVisible`, and `ApproximateNumberOfMessagesVisible` metrics are equal to 0 at least 1 minute after the producers stop sending messages (when the queue metadata reaches eventual consistency). For more information, see Available CloudWatch Metrics for Amazon SQS.

- Return messages as soon as they become available.

https://docs.aws.amazon.com/AWSSimpleQueueService/latest/SQSDeveloperGuide/sqs-long-polling.html

Option C is correct.



Amazon SQS Batch Actions

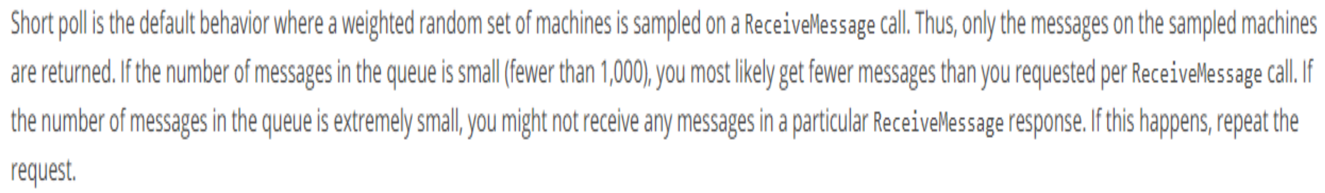
To reduce costs or manipulate up to 10 messages with a single action, you can use the following actions:

- `SendMessageBatch`
- `DeleteMessageBatch`
- `ChangeMessageVisibilityBatch`

You can take advantage of batch functionality using the Query API, or an AWS SDK that supports the Amazon SQS batch actions.

Option D is not correct.

Short polling does not guarantee a return of the message and you have to repeat the call until you receive the message. Which does not reduce any costs.



Short poll is the default behavior where a weighted random set of machines is sampled on a `ReceiveMessage` call. Thus, only the messages on the sampled machines are returned. If the number of messages in the queue is small (fewer than 1,000), you most likely get fewer messages than you requested per `ReceiveMessage` call. If the number of messages in the queue is extremely small, you might not receive any messages in a particular `ReceiveMessage` response. If this happens, repeat the request.

https://docs.aws.amazon.com/AWSSimpleQueueService/latest/APIReference/API_ReceiveMessage.ht

Ask our Experts

Rate this Question? ☺ ☹

Question 3 Incorrect

Domain : Other

Which of the following statements is not correct with respect to AWS SQS?

A. **Amazon SQS can trigger a lambda function.**

✓ B. To select the message to delete, use the ReceiptHandle of the message, not the MessageId which you receive when you send the message. ✗

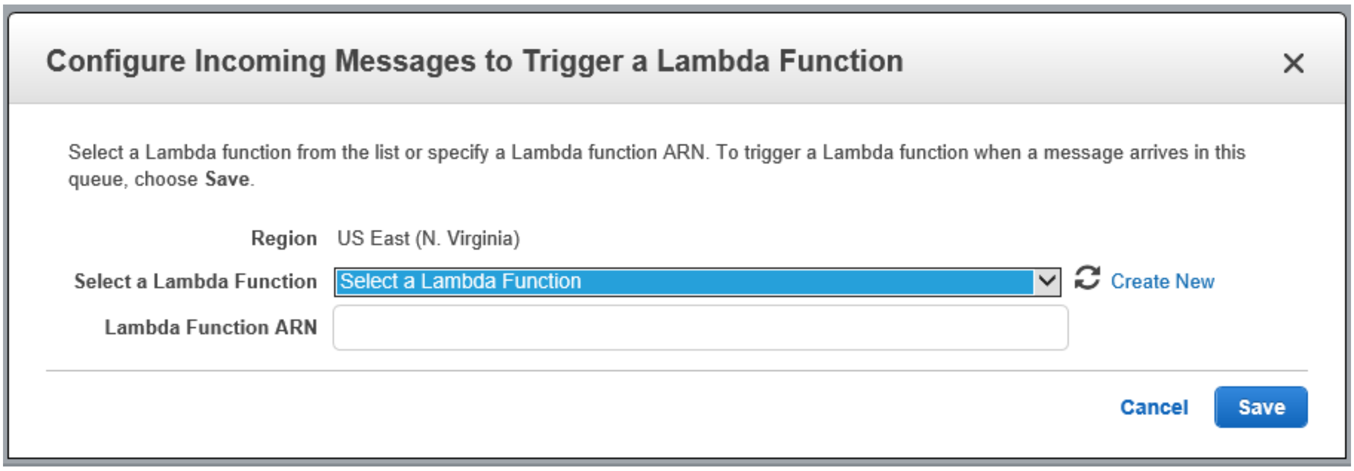
C. Use dead letter queues to isolate messages that can't be processed for later analysis.

D. All messages in Amazon SQS queue are encrypted by default. ✓

Explanation:

Answer: D

Option A is a correct statement.



Configure Incoming Messages to Trigger a Lambda Function

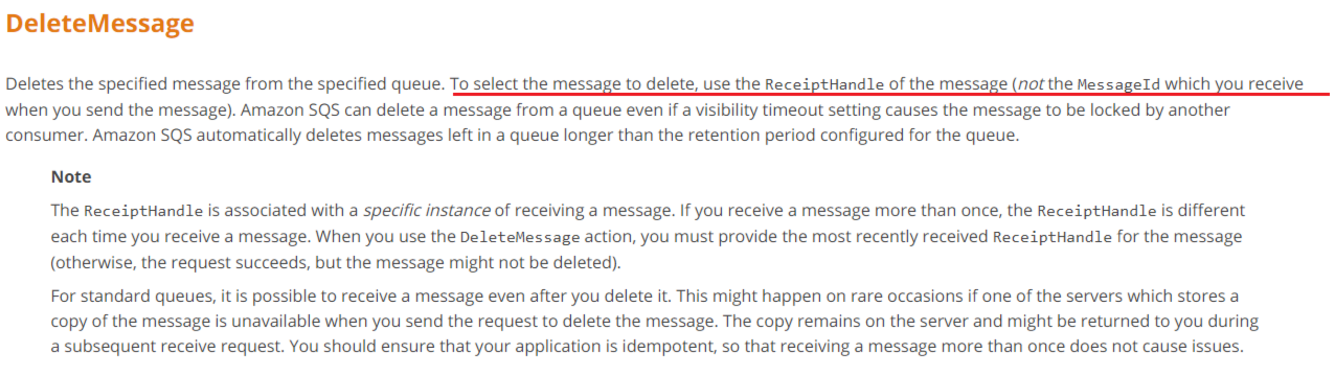
Select a Lambda function from the list or specify a Lambda function ARN. To trigger a Lambda function when a message arrives in this queue, choose **Save**.

Region: US East (N. Virginia)

Select a Lambda Function: Select a Lambda Function

Lambda Function ARN:

Option B is a correct statement.

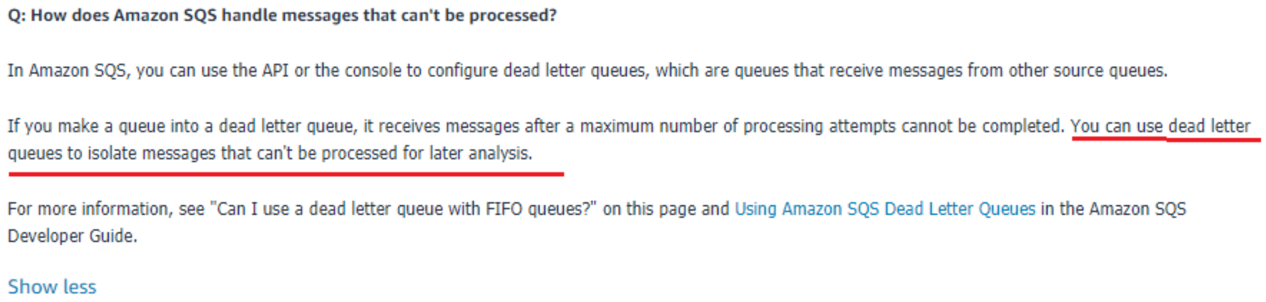


DeleteMessage

Deletes the specified message from the specified queue. To select the message to delete, use the `ReceiptHandle` of the message (*not* the `MessageId` which you receive when you send the message). Amazon SQS can delete a message from a queue even if a visibility timeout setting causes the message to be locked by another consumer. Amazon SQS automatically deletes messages left in a queue longer than the retention period configured for the queue.

> **Note**
>
> The `ReceiptHandle` is associated with a *specific instance* of receiving a message. If you receive a message more than once, the `ReceiptHandle` is different each time you receive a message. When you use the `DeleteMessage` action, you must provide the most recently received `ReceiptHandle` for the message (otherwise, the request succeeds, but the message might not be deleted).
>
> For standard queues, it is possible to receive a message even after you delete it. This might happen on rare occasions if one of the servers which stores a copy of the message is unavailable when you send the request to delete the message. The copy remains on the server and might be returned to you during a subsequent receive request. You should ensure that your application is idempotent, so that receiving a message more than once does not cause issues.

Option C is a correct statement.



Q: How does Amazon SQS handle messages that can't be processed?

In Amazon SQS, you can use the API or the console to configure dead letter queues, which are queues that receive messages from other source queues.

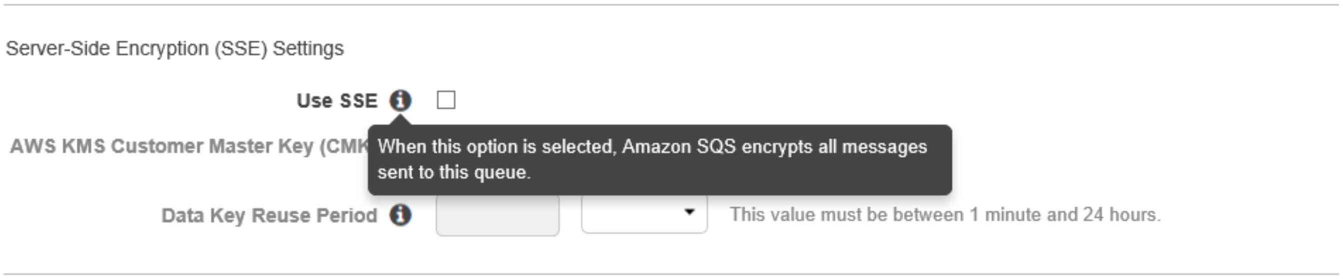
If you make a queue into a dead letter queue, it receives messages after a maximum number of processing attempts cannot be completed. You can use dead letter queues to isolate messages that can't be processed for later analysis.

For more information, see "Can I use a dead letter queue with FIFO queues?" on this page and Using Amazon SQS Dead Letter Queues in the Amazon SQS Developer Guide.

Show less

Option D is not a correct statement.

Amazon SQS does not encrypt messages by default. The option need to be selected by customer in order to enable encryption on the Queue messages.



Server-Side Encryption (SSE) Settings

Use SSE ℹ □

AWS KMS Customer Master Key (CMK

When this option is selected, Amazon SQS encrypts all messages sent to this queue.

Data Key Reuse Period ℹ This value must be between 1 minute and 24 hours.

https://aws.amazon.com/blogs/aws/new-server-side-encryption-for-amazon-simple-queue-service-sqs/

Ask our Experts

Rate this Question? ☺ ☹

Question 4 Incorrect

Domain : Other

Which of the following is not a feature of AWS Security Token Service?

A. STS enables you to request temporary, limited-privilege credentials.

✓ B. STS enables users to assume role. ✗

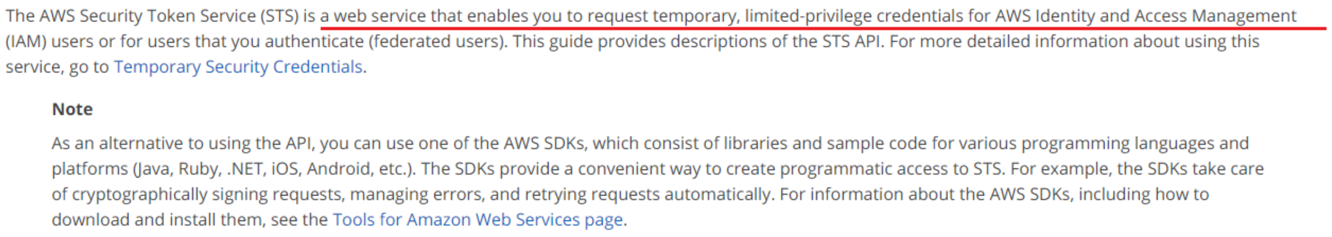
C. STS generates Git Credentials for IAM users. ✓

D. STS generates Federated Credentials for IAM users.

Explanation:

Answer: C

Option A is a correct statement.



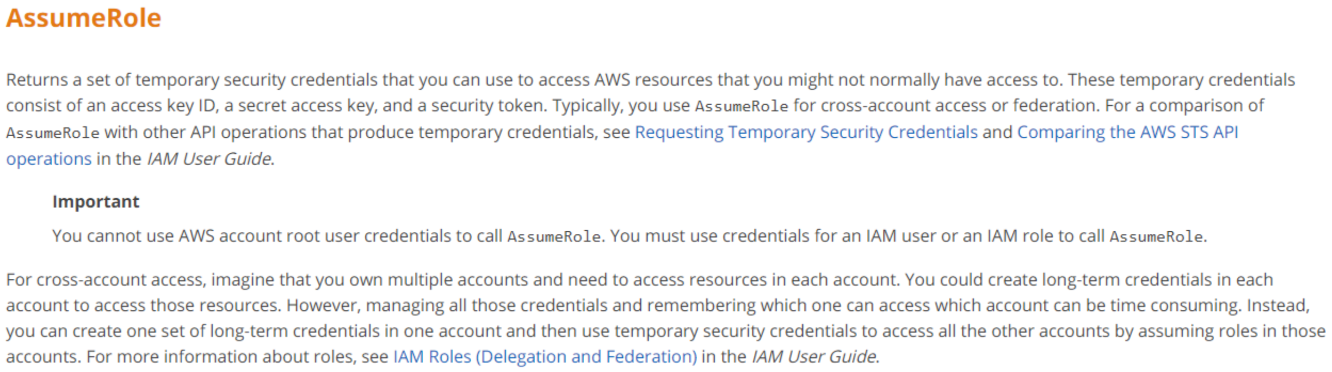
The AWS Security Token Service (STS) is a web service that enables you to request temporary, limited-privilege credentials for AWS Identity and Access Management (IAM) users or for users that you authenticate (federated users). This guide provides descriptions of the STS API. For more detailed information about using this service, go to Temporary Security Credentials.

Note

As an alternative to using the API, you can use one of the AWS SDKs, which consist of libraries and sample code for various programming languages and platforms (Java, Ruby, .NET, iOS, Android, etc.). The SDKs provide a convenient way to create programmatic access to STS. For example, the SDKs take care of cryptographically signing requests, managing errors, and retrying requests automatically. For information about the AWS SDKs, including how to download and install them, see the Tools for Amazon Web Services page.

Option B is a correct statement.

STS “AssumeRole” action will enable users to assume a role.



AssumeRole

Returns a set of temporary security credentials that you can use to access AWS resources that you might not normally have access to. These temporary credentials consist of an access key ID, a secret access key, and a security token. Typically, you use `AssumeRole` for cross-account access or federation. For a comparison of `AssumeRole` with other API operations that produce temporary credentials, see Requesting Temporary Security Credentials and Comparing the AWS STS API operations in the *IAM User Guide*.

Important

You cannot use AWS account root user credentials to call `AssumeRole`. You must use credentials for an IAM user or an IAM role to call `AssumeRole`.

For cross-account access, imagine that you own multiple accounts and need to access resources in each account. You could create long-term credentials in each account to access those resources. However, managing all those credentials and remembering which one can access which account can be time consuming. Instead, you can create one set of long-term credentials in one account and then use temporary security credentials to access all the other accounts by assuming roles in those accounts. For more information about roles, see IAM Roles (Delegation and Federation) in the *IAM User Guide*.

https://docs.aws.amazon.com/STS/latest/APIReference/API_AssumeRole.html

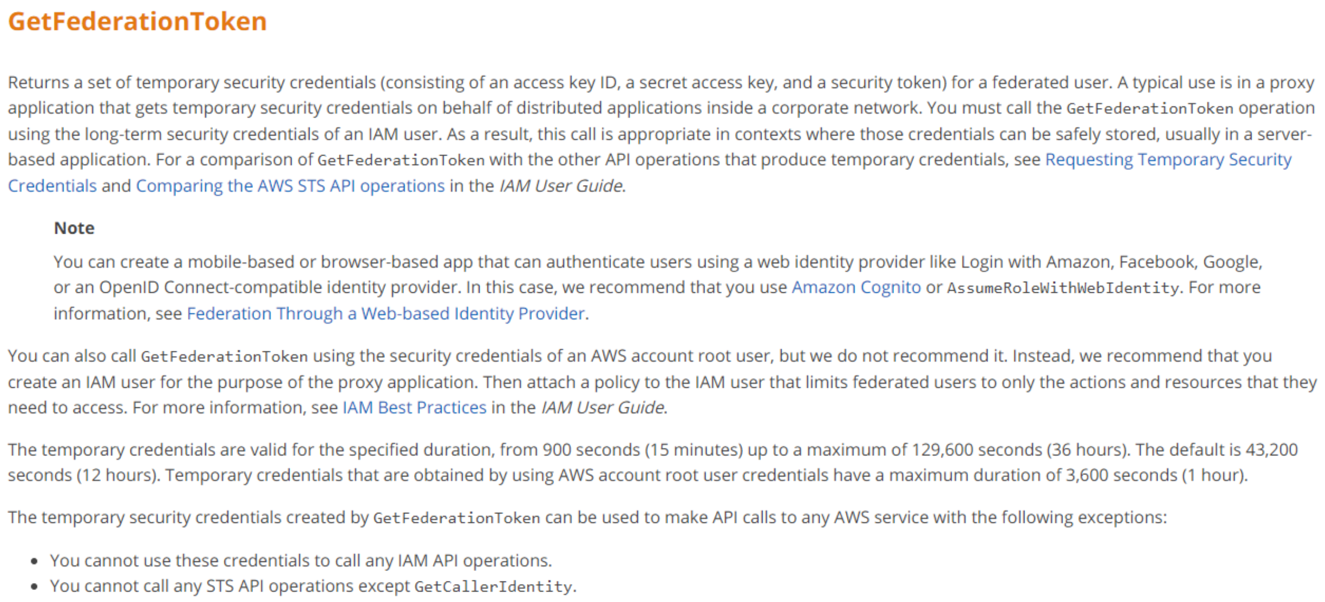
Option C is not a correct statement.

With Git credentials, you can generate a static user name and password in the Identity and Access Management (IAM) console that you can use to access AWS CodeCommit repositories from the command line, Git CLI, or any Git tool that supports HTTPS authentication.

This is not an action on AWS STS.

https://aws.amazon.com/blogs/devops/introducing-git-credentials-a-simple-way-to-connect-to-aws-codecommit-repositories-using-a-static-user-name-and-password/

Option D is a correct statement.



GetFederationToken

Returns a set of temporary security credentials (consisting of an access key ID, a secret access key, and a security token) for a federated user. A typical use is in a proxy application that gets temporary security credentials on behalf of distributed applications inside a corporate network. You must call the `GetFederationToken` operation using the long-term security credentials of an IAM user. As a result, this call is appropriate in contexts where those credentials can be safely stored, usually in a server-based application. For a comparison of `GetFederationToken` with the other API operations that produce temporary credentials, see Requesting Temporary Security Credentials and Comparing the AWS STS API operations in the *IAM User Guide*.

> **Note**
>
> You can create a mobile-based or browser-based app that can authenticate users using a web identity provider like Login with Amazon, Facebook, Google, or an OpenID Connect-compatible identity provider. In this case, we recommend that you use Amazon Cognito or `AssumeRoleWithWebIdentity`. For more information, see Federation Through a Web-based Identity Provider.

You can also call `GetFederationToken` using the security credentials of an AWS account root user, but we do not recommend it. Instead, we recommend that you create an IAM user for the purpose of the proxy application. Then attach a policy to the IAM user that limits federated users to only the actions and resources that they need to access. For more information, see IAM Best Practices in the *IAM User Guide*.

The temporary credentials are valid for the specified duration, from 900 seconds (15 minutes) up to a maximum of 129,600 seconds (36 hours). The default is 43,200 seconds (12 hours). Temporary credentials that are obtained by using AWS account root user credentials have a maximum duration of 3,600 seconds (1 hour).

The temporary security credentials created by `GetFederationToken` can be used to make API calls to any AWS service with the following exceptions:

- You cannot use these credentials to call any IAM API operations.
- You cannot call any STS API operations except `GetCallerIdentity`.

https://docs.aws.amazon.com/STS/latest/APIReference/API_GetFederationToken.html

Ask our Experts

Rate this Question? ☺ ☹

Question 5

Incorrect

Domain : Other

Your organization AWS Setup has an AWS S3 bucket which stores confidential documents which can be only downloaded by users authenticated and authorized via your application. You do not want to create IAM users for each of these users and as a best practice you have decided to generate AWS STS Federated User temporary credentials each time when a download request is made and then use the credentials to generate presigned URL and redirect user for download. However, when user is trying to access the presigned URL, they are getting Access Denied Error. What could be the reason?

✓ A. AWS STS service must be given access in S3 bucket ACL. ✗

B. IAM User used to generate Federated User credentials does not have access on S3 bucket. ✓

C. IAM Role used to generate Federated User credentials does not have access on S3 bucket.

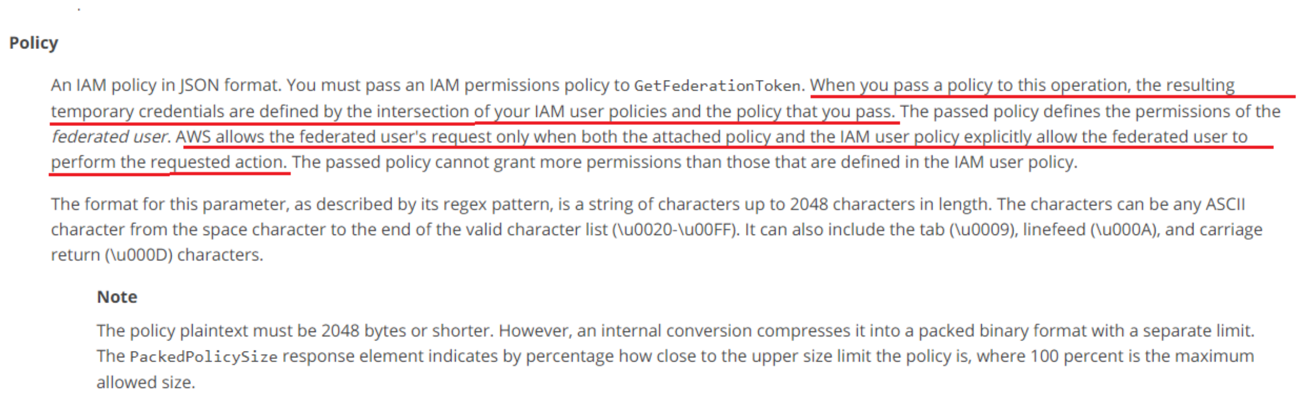
D. Your application must be whitelisted in AWS STS service to perform FederatedUser action.

Explanation:

Answer: B

Option A is not a correct statement.

Option B is correct.



Policy

An IAM policy in JSON format. You must pass an IAM permissions policy to `GetFederationToken`. When you pass a policy to this operation, the resulting temporary credentials are defined by the intersection of your IAM user policies and the policy that you pass. The passed policy defines the permissions of the *federated user*. AWS allows the federated user's request only when both the attached policy and the IAM user policy explicitly allow the federated user to perform the requested action. The passed policy cannot grant more permissions than those that are defined in the IAM user policy.

The format for this parameter, as described by its regex pattern, is a string of characters up to 2048 characters in length. The characters can be any ASCII character from the space character to the end of the valid character list (\u0020-\u00FF). It can also include the tab (\u0009), linefeed (\u000A), and carriage return (\u000D) characters.

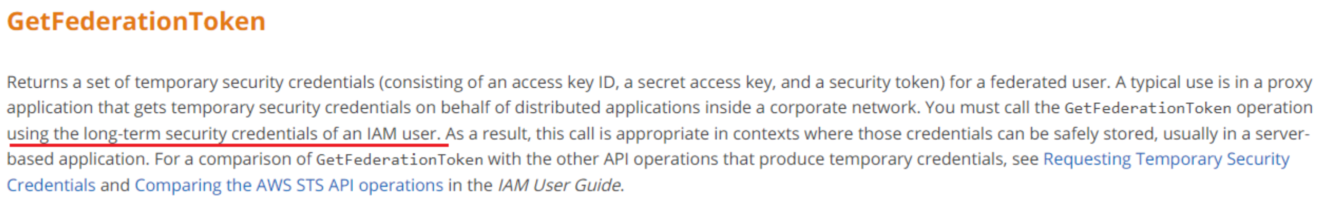
Note

The policy plaintext must be 2048 bytes or shorter. However, an internal conversion compresses it into a packed binary format with a separate limit. The `PackedPolicySize` response element indicates by percentage how close to the upper size limit the policy is, where 100 percent is the maximum allowed size.

https://docs.aws.amazon.com/STS/latest/APIReference/API_GetFederationToken.html

Option C is not correct.

You can generated FederatedUser credentials using an IAM User, not using an IAM Role.



GetFederationToken

Returns a set of temporary security credentials (consisting of an access key ID, a secret access key, and a security token) for a federated user. A typical use is in a proxy application that gets temporary security credentials on behalf of distributed applications inside a corporate network. You must call the `GetFederationToken` operation using the long-term security credentials of an IAM user. As a result, this call is appropriate in contexts where those credentials can be safely stored, usually in a server-based application. For a comparison of `GetFederationToken` with the other API operations that produce temporary credentials, see Requesting Temporary Security Credentials and Comparing the AWS STS API operations in the *IAM User Guide*.

Option D is not a correct statement.

Ask our Experts

Rate this Question? ☺ ☹

Question 6

Correct

Domain : Other

Your organization has an AWS setup and planning to build Single Sign On for users to authenticate with on-premise Microsoft Active Directory Federation Services (ADFS) and let users login to AWS console using AWS STS Enterprise Identity Federation. Which of the following service you need to call from AWS STS service after you authenticate with your on-premise?

✓ A. AssumeRoleWithSAML ✓

B. GetFederationToken

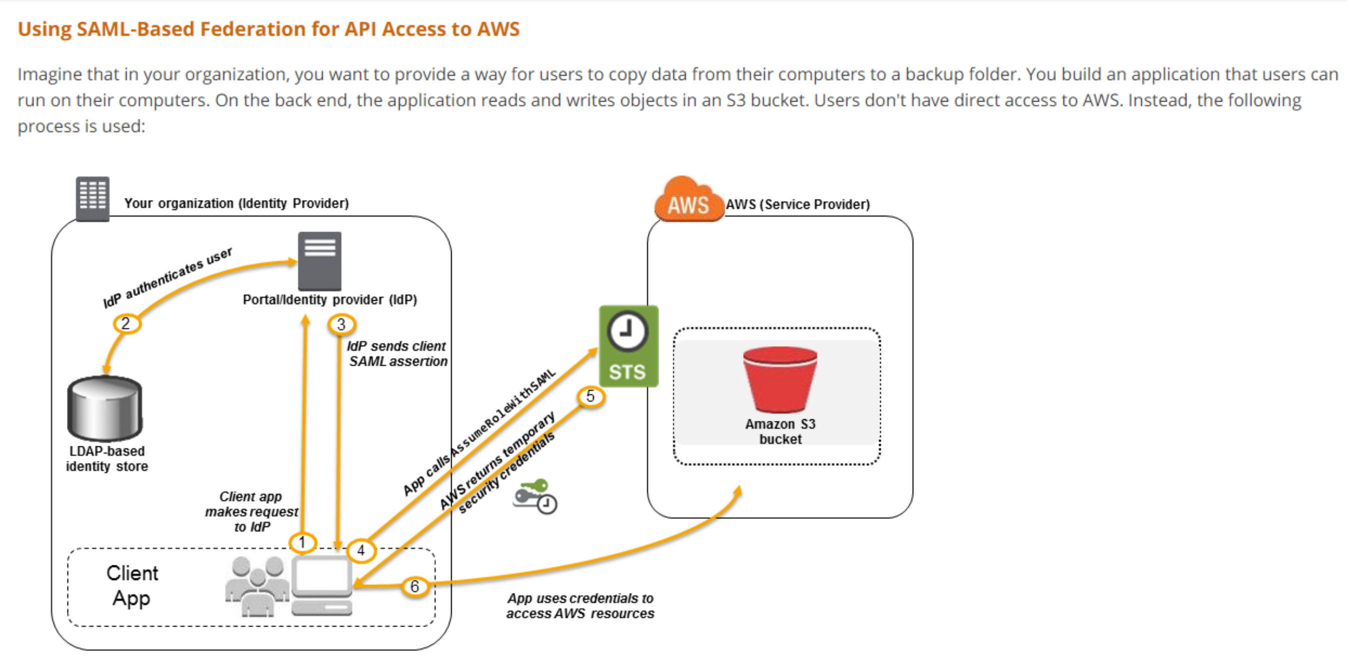
C. AssumeRoleWithWebIdentity

D. GetCallerIdentity

Explanation:

Answer: A

https://docs.aws.amazon.com/STS/latest/APIReference/API_AssumeRoleWithSAML.html



Using SAML-Based Federation for API Access to AWS

Imagine that in your organization, you want to provide a way for users to copy data from their computers to a backup folder. You build an application that users can run on their computers. On the back end, the application reads and writes objects in an S3 bucket. Users don't have direct access to AWS. Instead, the following process is used:

Your organization (Identity Provider)
AWS
AWS (Service Provider)
IdP authenticates user
2
Portal/Identity provider (IdP)
3
IdP sends client SAML assertion
LDAP-based identity store
STS
5
Amazon S3 bucket
App calls AssumeRoleWithSAML
AWS returns temporary security credentials
Client app makes request to IdP
1
4
Client App
6
App uses credentials to access AWS resources

https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles_providers_saml.html

Ask our Experts

Rate this Question? ☺ ☹

Question 7 Incorrect

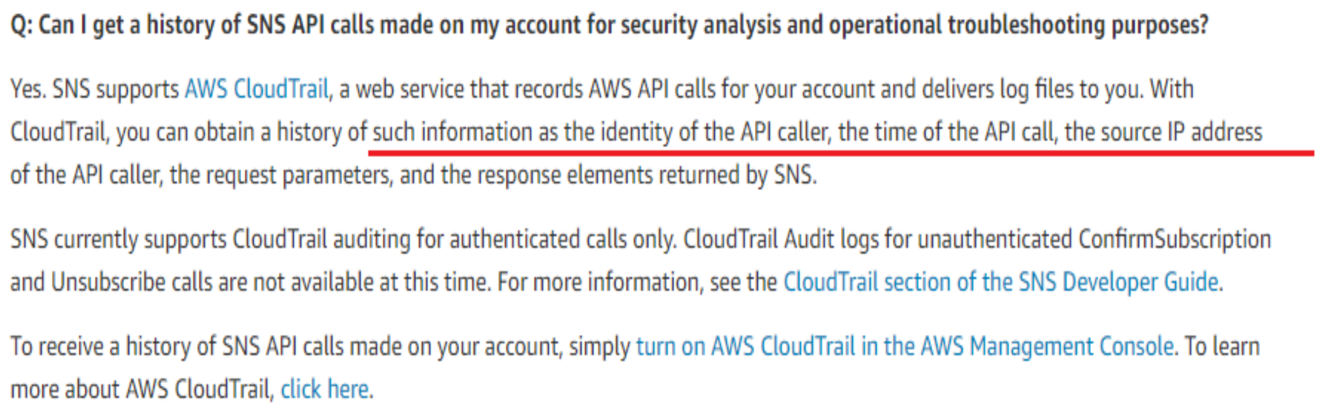
Domain : Other

Your organization has an AWS Setup. Your recent monthly bill has shown some increase in SNS billing. Your management wants to find out more about the incoming API calls. They want you to identify the requests made to AWS SNS on who made the request and the source IP address of the user who made the requests. How would you find out?

- ✓ A. **Enable SNS logging to S3 bucket.** ✗
- B. **Enable X-ray logging for SNS.**
- C. **Enable CloudTrail logging for SNS.** ✓
- D. **Enable CloudWatch logging for SNS.**

Explanation:

Answer: C



Q: Can I get a history of SNS API calls made on my account for security analysis and operational troubleshooting purposes?

Yes. SNS supports AWS CloudTrail, a web service that records AWS API calls for your account and delivers log files to you. With CloudTrail, you can obtain a history of such information as the identity of the API caller, the time of the API call, the source IP address of the API caller, the request parameters, and the response elements returned by SNS.

SNS currently supports CloudTrail auditing for authenticated calls only. CloudTrail Audit logs for unauthenticated ConfirmSubscription and Unsubscribe calls are not available at this time. For more information, see the CloudTrail section of the SNS Developer Guide.

To receive a history of SNS API calls made on your account, simply turn on AWS CloudTrail in the AWS Management Console. To learn more about AWS CloudTrail, click here.

Options A and B are not correct.

Option D is not correct. Amazon SNS and CloudWatch are integrated so you can collect, view, and analyze metrics for every active Amazon SNS notification. Once you have configured CloudWatch for Amazon SNS, you can gain better insight into the performance of your Amazon SNS topics, push notifications, and SMS deliveries.

But it does not provide information on the API calls made to SNS.

Ask our Experts

Rate this Question? ☺ ☹

Question 8

Correct

Domain : Other

Which of the following are the available options while configuring AWS SNS? (Choose 3 options)

✓ A. AWS Lambda ✓

✓ B. AWS SQS ✓

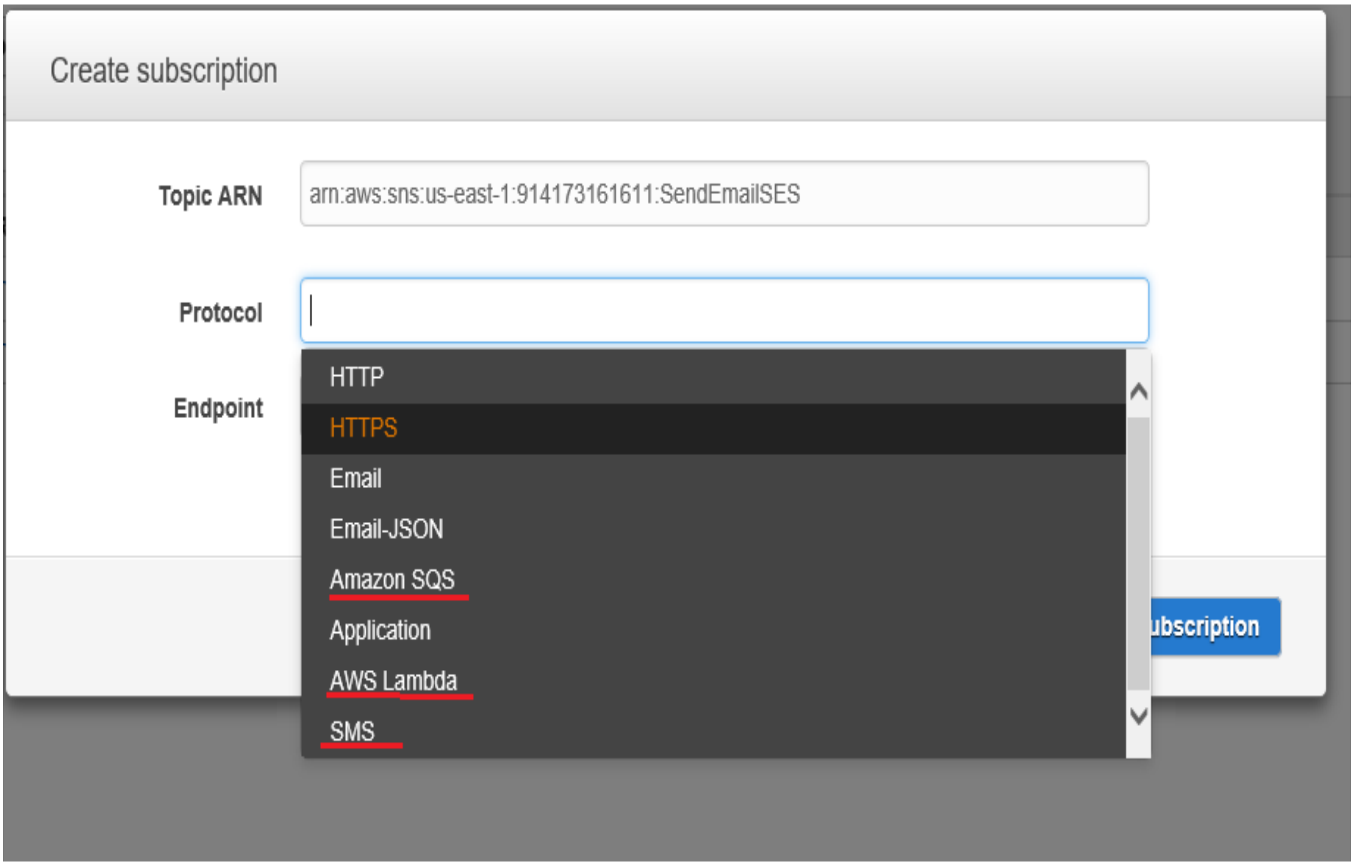
✓ C. SMS ✓

D. Email-XML

E. AWS MQ

Explanation:

Answer: A, B, C



Create subscription

Topic ARN: arn:aws:sns:us-east-1:914173161611:SendEmailSES

Protocol:

Endpoint

Q: What are the different delivery formats/transports for receiving notifications?

In order for customers to have broad flexibility of delivery mechanisms, Amazon SNS supports notifications over multiple transport protocols. Customers can select one the following transports as part of the subscription requests:

- "HTTP", "HTTPS" – Subscribers specify a URL as part of the subscription registration; notifications will be delivered through an HTTP POST to the specified URL.
- "Email", "Email-JSON" – Messages are sent to registered addresses as email. Email-JSON sends notifications as a JSON object, while Email sends text-based email.
- "SQS" – Users can specify an SQS standard queue as the endpoint; Amazon SNS will enqueue a notification message to the specified queue (which subscribers can then process using SQS APIs such as ReceiveMessage, DeleteMessage, etc.). Note that FIFO queues are not currently supported.
- "SMS" – Messages are sent to registered phone numbers as SMS text messages.

Ask our Experts

Rate this Question? ☺ ☹

Question 9

Incorrect

Domain : Other

You are an architect in your company and you have configured an SNS topic to send emails to a group of users regarding the CloudWatch alarms on the resource usages and outages. You were requested by your head of department to exclude him from those alarms except for critical system outages. How efficiently can you achieve this?

A. Create a new topic and and subscribe only head of department email address. Create new CloudWatch alarm only for critical outages and send messages to new Topic.

✓ **B. Configure another option on AWS CloudWatch alarm to send a direct email to head of department.** ✗

C. Add filter policy to head of department subscription. ✓

D. For head of department subscription, select AWS Lambda function which contains code to identify critical system outages and send email using AWS SES.

Explanation:

Answer: C

Option A is not correct.

Although it looks correct, it is not an efficient solution.

Option B is not correct. There is no such option on AWS CloudWatch alarms.

Option C is correct.

########

Amazon SNS Message Filtering

By default, a subscriber of an Amazon SNS topic receives every message published to the topic. To receive only a subset of the messages, a subscriber assigns a *filter policy* to the topic subscription.

A filter policy is a simple JSON object. The policy contains attributes that define which messages the subscriber receives. When you publish a message to a topic, Amazon SNS compares the message attributes to the attributes in the filter policy for each of the topic's subscriptions. If there is a match between the attributes, Amazon SNS sends the message to the subscriber. Otherwise, Amazon SNS skips the subscriber without sending the message to it. If a subscription lacks a filter policy, the subscription receives every message published to its topic.

With filter policies, you can simplify your usage of Amazon SNS by consolidating your message filtering criteria into your topic subscriptions. With this consolidation, you can offload the message filtering logic from subscribers and the message routing logic from publishers. Therefore, you don't need to filter messages by creating a separate topic for each filtering condition. Instead, you can use a single topic, and you can differentiate your messages with attributes. Each subscriber receives and processes only those messages accepted by its filter policy.

For example, you could use a single topic to publish all messages generated by transactions from your online retail site. To each message, you could assign an attribute that indicates the type of transaction, such as `order_placed`, `order_cancelled`, or `order_declined`. By creating subscriptions with filter policies, you can route each message to the queue that is meant to process the message's transaction type.

For a tutorial demonstrating how to implement message filtering with the AWS Management Console, see Filter Messages Published to Topics. This tutorial shows how to apply filter policies to route messages to separate Amazon SQS queues.

########

Option D is not correct.

Although it looks correct, it is not an efficient solution.

Ask our Experts

Rate this Question? ☺ ☹

Question 10 Incorrect

Domain : Other

Which of the following is not an item of message attribute in AWS SNS?

A. Name

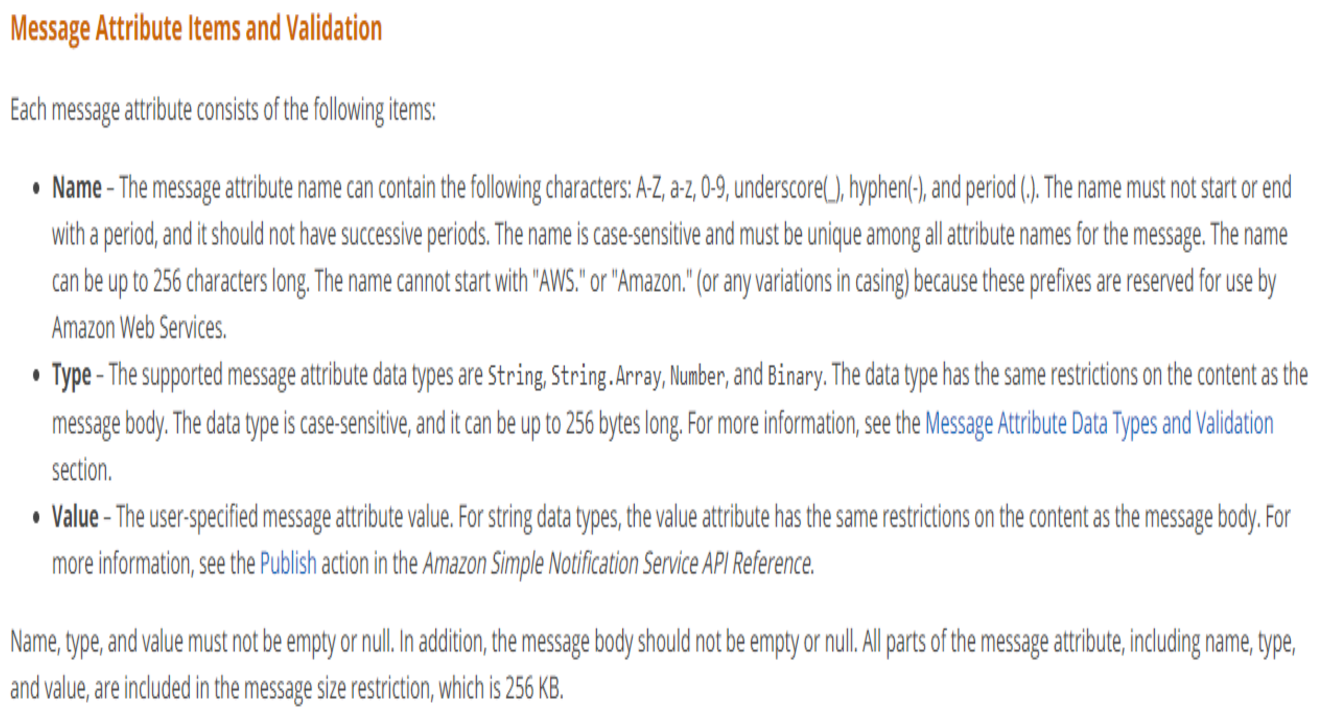
B. Type

✓ C. Value ✗

D. MessageID ✓

Explanation:

Answer: D

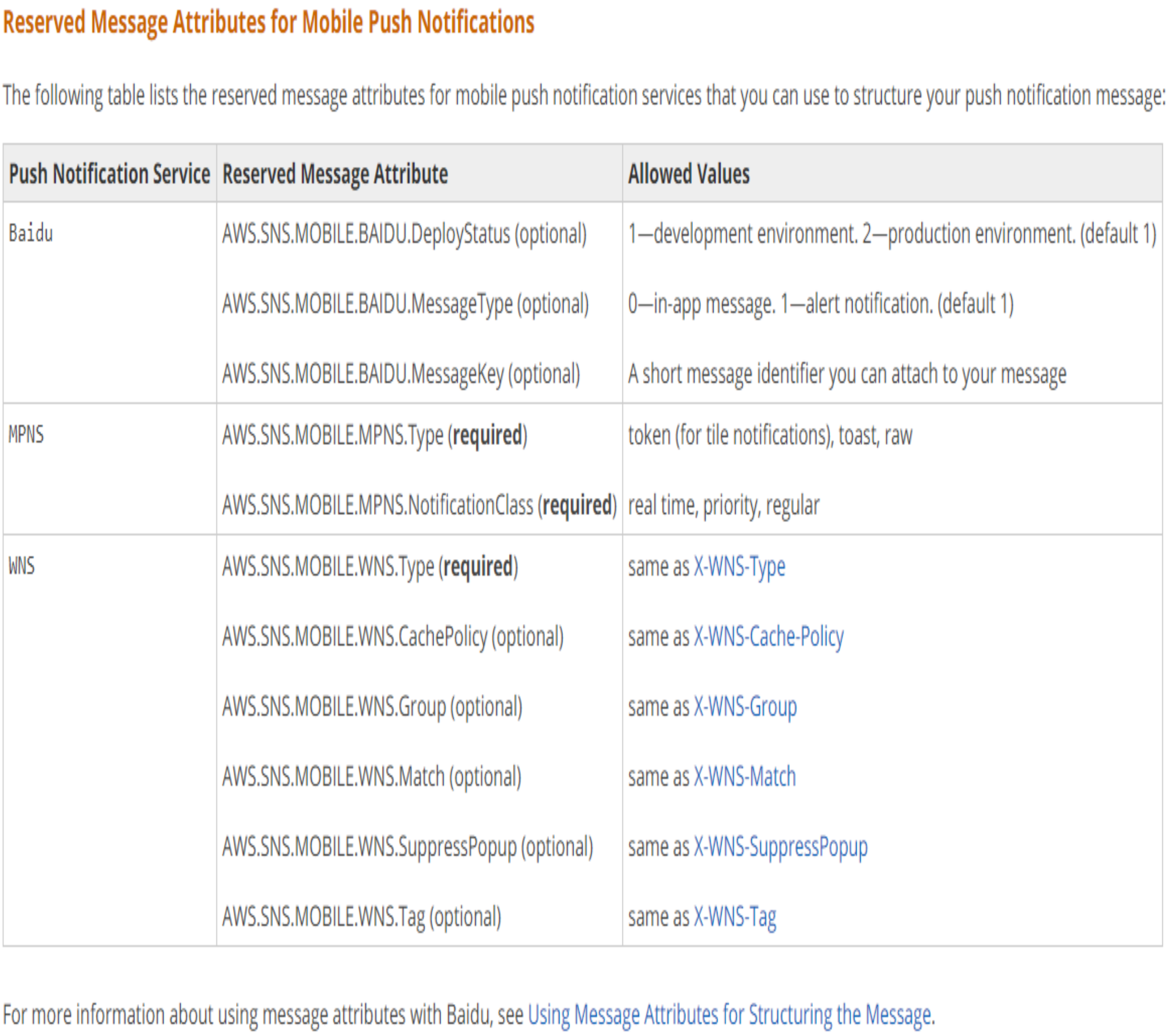


Message Attribute Items and Validation

Each message attribute consists of the following items:

- **Name** - The message attribute name can contain the following characters: A-Z, a-z, 0-9, underscore(_), hyphen(-), and period (.). The name must not start or end with a period, and it should not have successive periods. The name is case-sensitive and must be unique among all attribute names for the message. The name can be up to 256 characters long. The name cannot start with "AWS." or "Amazon." (or any variations in casing) because these prefixes are reserved for use by Amazon Web Services.
- **Type** - The supported message attribute data types are `String`, `String.Array`, `Number`, and `Binary`. The data type has the same restrictions on the content as the message body. The data type is case-sensitive, and it can be up to 256 bytes long. For more information, see the Message Attribute Data Types and Validation section.
- **Value** - The user-specified message attribute value. For string data types, the value attribute has the same restrictions on the content as the message body. For more information, see the Publish action in the *Amazon Simple Notification Service API Reference.*

Name, type, and value must not be empty or null. In addition, the message body should not be empty or null. All parts of the message attribute, including name, type, and value, are included in the message size restriction, which is 256 KB.



Reserved Message Attributes for Mobile Push Notifications

The following table lists the reserved message attributes for mobile push notification services that you can use to structure your push notification message:

Push Notification Service	Reserved Message Attribute	Allowed Values
Baidu	AWS.SNS.MOBILE.BAIDU.DeployStatus (optional)	1—development environment. 2—production environment. (default 1)
	AWS.SNS.MOBILE.BAIDU.MessageType (optional)	0—in-app message. 1—alert notification. (default 1)
	AWS.SNS.MOBILE.BAIDU.MessageKey (optional)	A short message identifier you can attach to your message
MPNS	AWS.SNS.MOBILE.MPNS.Type (**required**)	token (for tile notifications), toast, raw
	AWS.SNS.MOBILE.MPNS.NotificationClass (**required**)	real time, priority, regular
WNS	AWS.SNS.MOBILE.WNS.Type (**required**)	same as X-WNS-Type
	AWS.SNS.MOBILE.WNS.CachePolicy (optional)	same as X-WNS-Cache-Policy
	AWS.SNS.MOBILE.WNS.Group (optional)	same as X-WNS-Group
	AWS.SNS.MOBILE.WNS.Match (optional)	same as X-WNS-Match
	AWS.SNS.MOBILE.WNS.SuppressPopup (optional)	same as X-WNS-SuppressPopup
	AWS.SNS.MOBILE.WNS.Tag (optional)	same as X-WNS-Tag

For more information about using message attributes with Baidu, see Using Message Attributes for Structuring the Message.

https://docs.aws.amazon.com/sns/latest/dg/SNSMessageAttributes.html

Ask our Experts

Rate this Question? ☺ ☹

Finish Review

Certification

Cloud Certification

Java Certification

PM Certification

Big Data Certification

Company

Support

Discussions

Blog

Business

Follow us

© Copyright 2020. Whizlabs Software Pvt. Ltd. All Right Reserved.