Amazon Simple Notification Service

API Reference

API Version 2010-03-31

aws

Amazon Simple Notification Service: API Reference

Table of Contents

Welcome

Amazon Simple Notification Service (Amazon SNS) is a web service that enables you to build distributed web-enabled applications. Applications can use Amazon SNS to easily push real-time notification messages to interested subscribers over multiple delivery protocols. For more information about this product see https://aws.amazon.com/sns. For detailed information about Amazon SNS features and their associated API calls, see the Amazon SNS Developer Guide.

We also provide SDKs that enable you to access Amazon SNS from your preferred programming language. The SDKs contain functionality that automatically takes care of tasks such as: cryptographically signing your service requests, retrying requests, and handling error responses. For a list of available SDKs, go to Tools for Amazon Web Services.

This document was last published on January 7, 2020.

Actions

The following actions are supported:

AddPermission

Adds a statement to a topic's access control policy, granting access for the specified AWS accounts to the specified actions.

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

ActionName.member.N

The action you want to allow for the specified principal(s).

Valid values: Any Amazon SNS action name, for example `Publish`.

Type: Array of strings

Required: Yes

AWSAccountId.member.N

The AWS account IDs of the users (principals) who will be given access to the specified actions. The users must have AWS accounts, but do not need to be signed up for this service.

Type: Array of strings

Required: Yes

Label

A unique identifier for the new policy statement.

Type: String

Required: Yes

TopicArn

The ARN of the topic whose access control policy you wish to modify.

Type: String

Required: Yes

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

NotFound

Indicates that the requested resource does not exist.

HTTP Status Code: 404

Example

The structure of `AUTHPARAMS` depends on the signature of the API request. For more information, see Examples of Signed Signature Version 4 Requests in the *Amazon Web Services General Reference*.

Example

Sample Request

```
https://sns.us-east-2.amazonaws.com/?Action=AddPermission
&TopicArn=arn%3Aaws%3Asns%3Aus-east-2%3A123456789012%3AMy-Test
&Label=NewPermission
&ActionName.member.1=Publish
&ActionName.member.2=GetTopicAttributes
&AWSAccountId.member.1=987654321000
&AWSAccountId.member.2=876543210000
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<AddPermissionResponse xmlns="https://sns.amazonaws.com/doc/2010-03-31/">
    <ResponseMetadata>
        <RequestId>6a213e4e-33a8-11df-9540-99d0768312d3</RequestId>
    </ResponseMetadata>
</AddPermissionResponse>
```

See Also

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

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

CheckIfPhoneNumberIsOptedOut

Accepts a phone number and indicates whether the phone holder has opted out of receiving SMS messages from your account. You cannot send SMS messages to a number that is opted out.

To resume sending messages, you can opt in the number by using the `OptInPhoneNumber` action.

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

phoneNumber

The phone number for which you want to check the opt out status.

Type: String

Required: Yes

Response Elements

The following element is returned by the service.

isOptedOut

Indicates whether the phone number is opted out:
- `true` – The phone number is opted out, meaning you cannot publish SMS messages to it.
- `false` – The phone number is opted in, meaning you can publish SMS messages to it.

Type: Boolean

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

Throttled

Indicates that the rate at which requests have been submitted for this action exceeds the limit for your account.

HTTP Status Code: 429

See Also

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

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

ConfirmSubscription

Verifies an endpoint owner's intent to receive messages by validating the token sent to the endpoint by an earlier `Subscribe` action. If the token is valid, the action creates a new subscription and returns its Amazon Resource Name (ARN). This call requires an AWS signature only when the `AuthenticateOnUnsubscribe` flag is set to "true".

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

AuthenticateOnUnsubscribe

Disallows unauthenticated unsubscribes of the subscription. If the value of this parameter is `true` and the request has an AWS signature, then only the topic owner and the subscription owner can unsubscribe the endpoint. The unsubscribe action requires AWS authentication.

Type: String

Required: No

Token

Short-lived token sent to an endpoint during the `Subscribe` action.

Type: String

Required: Yes

TopicArn

The ARN of the topic for which you wish to confirm a subscription.

Type: String

Required: Yes

Response Elements

The following element is returned by the service.

SubscriptionArn

The ARN of the created subscription.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

FilterPolicyLimitExceeded

Indicates that the number of filter polices in your AWS account exceeds the limit. To add more filter polices, submit an SNS Limit Increase case in the AWS Support Center.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

NotFound

Indicates that the requested resource does not exist.

HTTP Status Code: 404

SubscriptionLimitExceeded

Indicates that the customer already owns the maximum allowed number of subscriptions.

HTTP Status Code: 403

Example

The structure of `AUTHPARAMS` depends on the signature of the API request. For more information, see Examples of Signed Signature Version 4 Requests in the *Amazon Web Services General Reference*.

Example

Sample Request

```
https://sns.us-east-2.amazonaws.com/?Action=ConfirmSubscription
&TopicArn=arn%3aws%3sns%3us-east-2%3123456789012%3My-Topic
&Token=51b2ff3edb475b7d91550e0ab6edf0c1de2a34e6ebaf6c2262a001bcb7e051c43aa00022ceecce70bd2a67b2042da8d8
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<ConfirmSubscriptionResponse xmlns="https://sns.amazonaws.com/doc/2010-03-31/">
    <ConfirmSubscriptionResult>
        <SubscriptionArn>arn:aws:sns:us-east-2:123456789012:My-Topic:80289ba6-0fd4-4079-
afb4-ce8c8260f0ca</SubscriptionArn>
    </ConfirmSubscriptionResult>
    <ResponseMetadata>
        <RequestId>7a50221f-3774-11df-a9b7-05d48da6f042</RequestId>
    </ResponseMetadata>
</ConfirmSubscriptionResponse>
```

See Also

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

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

CreatePlatformApplication

Creates a platform application object for one of the supported push notification services, such as APNS and GCM (Firebase Cloud Messaging), to which devices and mobile apps may register. You must specify `PlatformPrincipal` and `PlatformCredential` attributes when using the `CreatePlatformApplication` action.

`PlatformPrincipal` and `PlatformCredential` are received from the notification service.

- For `ADM`, `PlatformPrincipal` is `client id` and `PlatformCredential` is `client secret`.
- For `Baidu`, `PlatformPrincipal` is `API key` and `PlatformCredential` is `secret key`.
- For `APNS` and `APNS_SANDBOX`, `PlatformPrincipal` is `SSL certificate` and `PlatformCredential` is `private key`.
- For `GCM` (Firebase Cloud Messaging), there is no `PlatformPrincipal` and the `PlatformCredential` is `API key`.
- For `MPNS`, `PlatformPrincipal` is `TLS certificate` and `PlatformCredential` is `private key`.
- For `WNS`, `PlatformPrincipal` is `Package Security Identifier` and `PlatformCredential` is `secret key`.

You can use the returned `PlatformApplicationArn` as an attribute for the `CreatePlatformEndpoint` action.

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

Attributes , Attributes.entry.N.key (key), Attributes.entry.N.value (value)

For a list of attributes, see SetPlatformApplicationAttributes

Type: String to string map

Required: Yes

Name

Application names must be made up of only uppercase and lowercase ASCII letters, numbers, underscores, hyphens, and periods, and must be between 1 and 256 characters long.

Type: String

Required: Yes

Platform

The following platforms are supported: ADM (Amazon Device Messaging), APNS (Apple Push Notification Service), APNS_SANDBOX, and GCM (Firebase Cloud Messaging).

Type: String

Required: Yes

Response Elements

The following element is returned by the service.

PlatformApplicationArn

PlatformApplicationArn is returned.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

Example

The structure of `AUTHPARAMS` depends on the signature of the API request. For more information, see
Examples of Signed Signature Version 4 Requests in the *Amazon Web Services General Reference*.

Example

Sample Request

```
https://sns.us-west-2.amazonaws.com/?Action=CreatePlatformApplication
&Name=gcmpushapp
&Platform=GCM
&Attributes.entry.1.key=PlatformCredential
&Attributes.entry.1.value=AIzaSyClE2lcV2zEKTLYYo645zfk2jhQPFeyxDo
&Attributes.entry.2.key=PlatformPrincipal
&Attributes.entry.2.value=There+is+no+principal+for+GCM
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<CreatePlatformApplicationResponse xmlns="https://sns.amazonaws.com/doc/2010-03-31/">
    <CreatePlatformApplicationResult>
        <PlatformApplicationArn>arn:aws:sns:us-west-2:123456789012:app/GCM/gcmpushapp</
PlatformApplicationArn>
    </CreatePlatformApplicationResult>
    <ResponseMetadata>
        <RequestId>b6f0e78b-e9d4-5a0e-b973-adc04e8a4ff9</RequestId>
    </ResponseMetadata>
```

```
</CreatePlatformApplicationResponse>
```

See Also

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

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

CreatePlatformEndpoint

Creates an endpoint for a device and mobile app on one of the supported push notification services, such as GCM (Firebase Cloud Messaging) and APNS. `CreatePlatformEndpoint` requires the `PlatformApplicationArn` that is returned from `CreatePlatformApplication`. You can use the returned `EndpointArn` to send a message to a mobile app or by the `Subscribe` action for subscription to a topic. The `CreatePlatformEndpoint` action is idempotent, so if the requester already owns an endpoint with the same device token and attributes, that endpoint's ARN is returned without creating a new endpoint. For more information, see Using Amazon SNS Mobile Push Notifications.

When using `CreatePlatformEndpoint` with Baidu, two attributes must be provided: ChannelId and UserId. The token field must also contain the ChannelId. For more information, see Creating an Amazon SNS Endpoint for Baidu.

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

Attributes , Attributes.entry.N.key (key), Attributes.entry.N.value (value)

For a list of attributes, see SetEndpointAttributes.

Type: String to string map

Required: No

CustomUserData

Arbitrary user data to associate with the endpoint. Amazon SNS does not use this data. The data must be in UTF-8 format and less than 2KB.

Type: String

Required: No

PlatformApplicationArn

PlatformApplicationArn returned from CreatePlatformApplication is used to create a an endpoint.

Type: String

Required: Yes

Token

Unique identifier created by the notification service for an app on a device. The specific name for Token will vary, depending on which notification service is being used. For example, when using APNS as the notification service, you need the device token. Alternatively, when using GCM (Firebase Cloud Messaging) or ADM, the device token equivalent is called the registration ID.

Type: String

Required: Yes

Response Elements

The following element is returned by the service.

EndpointArn

EndpointArn returned from CreateEndpoint action.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

NotFound

Indicates that the requested resource does not exist.

HTTP Status Code: 404

Example

The structure of `AUTHPARAMS` depends on the signature of the API request. For more information, see Examples of Signed Signature Version 4 Requests in the *Amazon Web Services General Reference*.

Example

Sample Request

```
https://sns.us-west-2.amazonaws.com/?Action=CreatePlatformEndpoint
&PlatformApplicationArn=arn%3Aaws%3Asns%3Aus-west-2%3A123456789012%3Aapp%2FGCM%2Fgcmpushapp
&Token=APA91bGi7fFachkC1xjlqT66VYEucGHochmf1VQAr9k...jsM0PKPxKhddCzx6paEsyay9Zn3D4wNUJb8m6HZrBEXAMPLE
&CustomUserData=UserId%3D27576823
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<CreatePlatformEndpointResponse xmlns="https://sns.amazonaws.com/doc/2010-03-31/">
    <CreatePlatformEndpointResult>
        <EndpointArn>arn:aws:sns:us-west-2:123456789012:endpoint/GCM/
gcmpushapp/5e3e9847-3183-3f18-a7e8-671c3a57d4b3</EndpointArn>
    </CreatePlatformEndpointResult>
    <ResponseMetadata>
```

```
        <RequestId>6613341d-3e15-53f7-bf3c-7e56994ba278</RequestId>
    </ResponseMetadata>
</CreatePlatformEndpointResponse>
```

See Also

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

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

CreateTopic

Creates a topic to which notifications can be published. Users can create at most 100,000 topics. For more information, see https://aws.amazon.com/sns. This action is idempotent, so if the requester already owns a topic with the specified name, that topic's ARN is returned without creating a new topic.

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

Attributes , Attributes.entry.N.key (key), Attributes.entry.N.value (value)

A map of attributes with their corresponding values.

The following lists the names, descriptions, and values of the special request parameters that the `CreateTopic` action uses:

- `DeliveryPolicy` – The policy that defines how Amazon SNS retries failed deliveries to HTTP/S endpoints.
- `DisplayName` – The display name to use for a topic with SMS subscriptions.
- `Policy` – The policy that defines who can access your topic. By default, only the topic owner can publish or subscribe to the topic.

The following attribute applies only to server-side-encryption:

- `KmsMasterKeyId` - The ID of an AWS-managed customer master key (CMK) for Amazon SNS or a custom CMK. For more information, see Key Terms. For more examples, see KeyId in the *AWS Key Management Service API Reference*.

Type: String to string map

Required: No

Name

The name of the topic you want to create.

Constraints: Topic names must be made up of only uppercase and lowercase ASCII letters, numbers, underscores, and hyphens, and must be between 1 and 256 characters long.

Type: String

Required: Yes

Tags.member.N

The list of tags to add to a new topic.

> **Note**
> To be able to tag a topic on creation, you must have the `sns:CreateTopic` and `sns:TagResource` permissions.

Type: Array of Tag (p. 95) objects

Required: No

Response Elements

The following element is returned by the service.

TopicArn

The Amazon Resource Name (ARN) assigned to the created topic.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

ConcurrentAccess

Can't perform multiple operations on a tag simultaneously. Perform the operations sequentially.

HTTP Status Code: 400

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

InvalidSecurity

The credential signature isn't valid. You must use an HTTPS endpoint and sign your request using Signature Version 4.

HTTP Status Code: 403

StaleTag

A tag has been added to a resource with the same ARN as a deleted resource. Wait a short while and then retry the operation.

HTTP Status Code: 400

TagLimitExceeded

Can't add more than 50 tags to a topic.

HTTP Status Code: 400

TagPolicy

The request doesn't comply with the IAM tag policy. Correct your request and then retry it.

HTTP Status Code: 400

TopicLimitExceeded

Indicates that the customer already owns the maximum allowed number of topics.

HTTP Status Code: 403

Example

The structure of `AUTHPARAMS` depends on the signature of the API request. For more information, see Examples of Signed Signature Version 4 Requests in the *Amazon Web Services General Reference*.

Example

Sample Request

```
https://sns.us-east-2.amazonaws.com/?Action=CreateTopic
&Name=My-Topic
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<CreateTopicResponse xmlns="https://sns.amazonaws.com/doc/2010-03-31/">
    <CreateTopicResult>
        <TopicArn>arn:aws:sns:us-east-2:123456789012:My-Topic</TopicArn>
    </CreateTopicResult>
    <ResponseMetadata>
        <RequestId>a8dec8b3-33a4-11df-8963-01868b7c937a</RequestId>
    </ResponseMetadata>
</CreateTopicResponse>
```

See Also

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

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

DeleteEndpoint

Deletes the endpoint for a device and mobile app from Amazon SNS. This action is idempotent. For more information, see Using Amazon SNS Mobile Push Notifications.

When you delete an endpoint that is also subscribed to a topic, then you must also unsubscribe the endpoint from the topic.

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

EndpointArn

EndpointArn of endpoint to delete.

Type: String

Required: Yes

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

Example

The structure of AUTHPARAMS depends on the signature of the API request. For more information, see Examples of Signed Signature Version 4 Requests in the *Amazon Web Services General Reference*.

Example

Sample Request

```
https://sns.us-west-2.amazonaws.com/?Action=DeleteEndpoint
&EndpointArn=arn%3Aaws%3Asns%3Aus-west-2%3A123456789012%3Aendpoint%2FGCM%2Fgcmpushapp
%2F5e3e9847-3183-3f18-a7e8-671c3a57d4b3
&Version=2010-03-31
```

```
&AUTHPARAMS
```

Sample Response

```
<DeleteEndpointResponse xmlns="https://sns.amazonaws.com/doc/2010-03-31/">
    <ResponseMetadata>
        <RequestId>c1d2b191-353c-5a5f-8969-fbdd3900afa8</RequestId>
    </ResponseMetadata>
</DeleteEndpointResponse>
```

See Also

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

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

DeletePlatformApplication

Deletes a platform application object for one of the supported push notification services, such as APNS and GCM (Firebase Cloud Messaging). For more information, see Using Amazon SNS Mobile Push Notifications.

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

PlatformApplicationArn

PlatformApplicationArn of platform application object to delete.

Type: String

Required: Yes

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

Example

The structure of `AUTHPARAMS` depends on the signature of the API request. For more information, see Examples of Signed Signature Version 4 Requests in the *Amazon Web Services General Reference*.

Example

Sample Request

```
https://sns.us-west-2.amazonaws.com/?Action=DeletePlatformApplication
&PlatformApplicationArn=arn%3Aaws%3Asns%3Aus-west-2%3A123456789012%3Aapp%2FGCM%2Fgcmpushapp
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<DeletePlatformApplicationResponse xmlns="https://sns.amazonaws.com/doc/2010-03-31/">
    <ResponseMetadata>
        <RequestId>097dac18-7a77-5823-a8dd-e65476dcb037</RequestId>
    </ResponseMetadata>
</DeletePlatformApplicationResponse>
```

See Also

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

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

DeleteTopic

Deletes a topic and all its subscriptions. Deleting a topic might prevent some messages previously sent to the topic from being delivered to subscribers. This action is idempotent, so deleting a topic that does not exist does not result in an error.

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

TopicArn

> The ARN of the topic you want to delete.
>
> Type: String
>
> Required: Yes

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

> Indicates that the user has been denied access to the requested resource.
>
> HTTP Status Code: 403

ConcurrentAccess

> Can't perform multiple operations on a tag simultaneously. Perform the operations sequentially.
>
> HTTP Status Code: 400

InternalError

> Indicates an internal service error.
>
> HTTP Status Code: 500

InvalidParameter

> Indicates that a request parameter does not comply with the associated constraints.
>
> HTTP Status Code: 400

NotFound

> Indicates that the requested resource does not exist.
>
> HTTP Status Code: 404

StaleTag

> A tag has been added to a resource with the same ARN as a deleted resource. Wait a short while and then retry the operation.
>
> HTTP Status Code: 400

TagPolicy

> The request doesn't comply with the IAM tag policy. Correct your request and then retry it.

HTTP Status Code: 400

Example

The structure of AUTHPARAMS depends on the signature of the API request. For more information, see Examples of Signed Signature Version 4 Requests in the *Amazon Web Services General Reference*.

Example

Sample Request

```
https://sns.us-east-2.amazonaws.com/?Action=DeleteTopic
&TopicArn=arn%3Aaws%3Asns%3Aus-east-2%3A123456789012%3AMy-Topic
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<DeleteTopicResponse xmlns="https://sns.amazonaws.com/doc/2010-03-31/">
    <ResponseMetadata>
        <RequestId>f3aa9ac9-3c3d-11df-8235-9dab105e9c32</RequestId>
    </ResponseMetadata>
</DeleteTopicResponse>
```

See Also

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

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

GetEndpointAttributes

Retrieves the endpoint attributes for a device on one of the supported push notification services, such as GCM (Firebase Cloud Messaging) and APNS. For more information, see Using Amazon SNS Mobile Push Notifications.

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

EndpointArn

EndpointArn for GetEndpointAttributes input.

Type: String

Required: Yes

Response Elements

The following element is returned by the service.

Attributes , Attributes.entry.N.key (key), Attributes.entry.N.value (value)

Attributes include the following:
- `CustomUserData` – arbitrary user data to associate with the endpoint. Amazon SNS does not use this data. The data must be in UTF-8 format and less than 2KB.
- `Enabled` – flag that enables/disables delivery to the endpoint. Amazon SNS will set this to false when a notification service indicates to Amazon SNS that the endpoint is invalid. Users can set it back to true, typically after updating Token.
- `Token` – device token, also referred to as a registration id, for an app and mobile device. This is returned from the notification service when an app and mobile device are registered with the notification service.

    Note
    The device token for the iOS platform is returned in lowercase.

Type: String to string map

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

NotFound

Indicates that the requested resource does not exist.

HTTP Status Code: 404

Example

The structure of `AUTHPARAMS` depends on the signature of the API request. For more information, see Examples of Signed Signature Version 4 Requests in the *Amazon Web Services General Reference*.

Example

Sample Request

```
https://sns.us-west-2.amazonaws.com/?Action=GetEndpointAttributes
&EndpointArn=arn%3Aaws%3Asns%3Aus-west-2%3A123456789012%3Aendpoint%2FGCM%2Fgcmpushapp
%2F5e3e9847-3183-3f18-a7e8-671c3a57d4b3
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<GetEndpointAttributesResponse xmlns="https://sns.amazonaws.com/doc/2010-03-31/">
    <GetEndpointAttributesResult>
        <Attributes>
            <entry>
                <key>Enabled</key>
                <value>true</value>
            </entry>
            <entry>
                <key>CustomUserData</key>
                <value>UserId=01234567</value>
            </entry>
            <entry>
                <key>Token</key>

 <value>APA91bGi7fFachkC1xjlqT66VYEucGHochmf1VQAr9k...jsM0PKPxKhddCzx6paEsyay9Zn3D4wNUJb8m6HZrBEXAMPLE<
value>
            </entry>
        </Attributes>
    </GetEndpointAttributesResult>
    <ResponseMetadata>
        <RequestId>6c725a19-a142-5b77-94f9-1055a9ea04e7</RequestId>
    </ResponseMetadata>
</GetEndpointAttributesResponse>
```

See Also

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

- AWS Command Line Interface

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

GetPlatformApplicationAttributes

Retrieves the attributes of the platform application object for the supported push notification services, such as APNS and GCM (Firebase Cloud Messaging). For more information, see Using Amazon SNS Mobile Push Notifications.

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

PlatformApplicationArn

PlatformApplicationArn for GetPlatformApplicationAttributesInput.

Type: String

Required: Yes

Response Elements

The following element is returned by the service.

Attributes , Attributes.entry.N.key (key), Attributes.entry.N.value (value)

Attributes include the following:
- `EventEndpointCreated` – Topic ARN to which EndpointCreated event notifications should be sent.
- `EventEndpointDeleted` – Topic ARN to which EndpointDeleted event notifications should be sent.
- `EventEndpointUpdated` – Topic ARN to which EndpointUpdate event notifications should be sent.
- `EventDeliveryFailure` – Topic ARN to which DeliveryFailure event notifications should be sent upon Direct Publish delivery failure (permanent) to one of the application's endpoints.

Type: String to string map

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

NotFound

Indicates that the requested resource does not exist.

HTTP Status Code: 404

Example

The structure of `AUTHPARAMS` depends on the signature of the API request. For more information, see
Examples of Signed Signature Version 4 Requests in the *Amazon Web Services General Reference*.

Example

Sample Request

```
https://sns.us-west-2.amazonaws.com/?Action=GetPlatformApplicationAttributes
&PlatformApplicationArn=arn%3Aaws%3Asns%3Aus-west-2%3A123456789012%3Aapp%2FGCM%2Fgcmpushapp
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<GetPlatformApplicationAttributesResponse xmlns="https://sns.amazonaws.com/
doc/2010-03-31/">
    <GetPlatformApplicationAttributesResult>
        <Attributes>
            <entry>
                <key>AllowEndpointPolicies</key>
                <value>false</value>
            </entry>
        </Attributes>
    </GetPlatformApplicationAttributesResult>
    <ResponseMetadata>
        <RequestId>74848df2-87f6-55ed-890c-c7be80442462</RequestId>
    </ResponseMetadata>
</GetPlatformApplicationAttributesResponse>
```

See Also

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

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

GetSMSAttributes

Returns the settings for sending SMS messages from your account.

These settings are set with the `SetSMSAttributes` action.

Request Parameters

For information about the parameters that are common to all actions, see Common
Parameters (p. 97).

attributes.member.N

A list of the individual attribute names, such as `MonthlySpendLimit`, for which you want values.

For all attribute names, see SetSMSAttributes.

If you don't use this parameter, Amazon SNS returns all SMS attributes.

Type: Array of strings

Required: No

Response Elements

The following element is returned by the service.

attributes , attributes.entry.N.key (key), attributes.entry.N.value (value)

The SMS attribute names and their values.

Type: String to string map

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

Throttled

Indicates that the rate at which requests have been submitted for this action exceeds the limit for
your account.

HTTP Status Code: 429

See Also

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

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

GetSubscriptionAttributes

Returns all of the properties of a subscription.

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

SubscriptionArn

The ARN of the subscription whose properties you want to get.

Type: String

Required: Yes

Response Elements

The following element is returned by the service.

Attributes , Attributes.entry.N.key (key), Attributes.entry.N.value (value)

A map of the subscription's attributes. Attributes in this map include the following:

- `ConfirmationWasAuthenticated` – `true` if the subscription confirmation request was authenticated.
- `DeliveryPolicy` – The JSON serialization of the subscription's delivery policy.
- `EffectiveDeliveryPolicy` – The JSON serialization of the effective delivery policy that takes into account the topic delivery policy and account system defaults.
- `FilterPolicy` – The filter policy JSON that is assigned to the subscription. For more information, see Amazon SNS Message Filtering in the *Amazon SNS Developer Guide*.
- `Owner` – The AWS account ID of the subscription's owner.
- `PendingConfirmation` – `true` if the subscription hasn't been confirmed. To confirm a pending subscription, call the `ConfirmSubscription` action with a confirmation token.
- `RawMessageDelivery` – `true` if raw message delivery is enabled for the subscription. Raw messages are free of JSON formatting and can be sent to HTTP/S and Amazon SQS endpoints.
- `RedrivePolicy` – When specified, sends undeliverable messages to the specified Amazon SQS dead-letter queue. Messages that can't be delivered due to client errors (for example, when the subscribed endpoint is unreachable) or server errors (for example, when the service that powers the subscribed endpoint becomes unavailable) are held in the dead-letter queue for further analysis or reprocessing.
- `SubscriptionArn` – The subscription's ARN.
- `TopicArn` – The topic ARN that the subscription is associated with.

Type: String to string map

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

NotFound

Indicates that the requested resource does not exist.

HTTP Status Code: 404

Example

The structure of `AUTHPARAMS` depends on the signature of the API request. For more information, see Examples of Signed Signature Version 4 Requests in the *Amazon Web Services General Reference*.

Example

Sample Request

```
https://sns.us-east-2.amazonaws.com/?Action=GetSubscriptionAttributes
&SubscriptionArn=arn%3Aaws%3Asns%3Aus-east-2%3A123456789012%3AMy-Topic
%3A80289ba6-0fd4-4079-afb4-ce8c8260f0ca
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<GetSubscriptionAttributesResponse xmlns="https://sns.amazonaws.com/doc/2010-03-31/">
    <GetSubscriptionAttributesResult>
        <Attributes>
            <entry>
                <key>Owner</key>
                <value>123456789012</value>
            </entry>
            <entry>
                <key>DeliveryPolicy</key>
                <value>{&quot;healthyRetryPolicy&quot;:{&quot;numRetries&quot;:10}}</value>
            </entry>
            <entry>
                <key>SubscriptionArn</key>
                <value>arn:aws:sns:us-east-2:123456789012:My-Topic:80289ba6-0fd4-4079-afb4-
ce8c8260f0ca</value>
            </entry>
        </Attributes>
    </GetSubscriptionAttributesResult>
    <ResponseMetadata>
        <RequestId>057f074c-33a7-11df-9540-99d0768312d3</RequestId>
    </ResponseMetadata>
</GetTopicAttributesResponse>
```

See Also

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

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

GetTopicAttributes

Returns all of the properties of a topic. Topic properties returned might differ based on the authorization of the user.

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

TopicArn

The ARN of the topic whose properties you want to get.

Type: String

Required: Yes

Response Elements

The following element is returned by the service.

Attributes , Attributes.entry.N.key (key), Attributes.entry.N.value (value)

A map of the topic's attributes. Attributes in this map include the following:
- `DeliveryPolicy` – The JSON serialization of the topic's delivery policy.
- `DisplayName` – The human-readable name used in the `From` field for notifications to `email` and `email-json` endpoints.
- `Owner` – The AWS account ID of the topic's owner.
- `Policy` – The JSON serialization of the topic's access control policy.
- `SubscriptionsConfirmed` – The number of confirmed subscriptions for the topic.
- `SubscriptionsDeleted` – The number of deleted subscriptions for the topic.
- `SubscriptionsPending` – The number of subscriptions pending confirmation for the topic.
- `TopicArn` – The topic's ARN.
- `EffectiveDeliveryPolicy` – Yhe JSON serialization of the effective delivery policy, taking system defaults into account.

The following attribute applies only to server-side-encryption:
- `KmsMasterKeyId` - The ID of an AWS-managed customer master key (CMK) for Amazon SNS or a custom CMK. For more information, see Key Terms. For more examples, see KeyId in the *AWS Key Management Service API Reference*.

Type: String to string map

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

InvalidSecurity

The credential signature isn't valid. You must use an HTTPS endpoint and sign your request using Signature Version 4.

HTTP Status Code: 403

NotFound

Indicates that the requested resource does not exist.

HTTP Status Code: 404

Example

The structure of `AUTHPARAMS` depends on the signature of the API request. For more information, see Examples of Signed Signature Version 4 Requests in the *Amazon Web Services General Reference*.

Example

Sample Request

```
https://sns.us-east-2.amazonaws.com/?Action=GetTopicAttributes
&TopicArn=arn%3Aaws%3Asns%3Aus-east-2%3A123456789012%3AMy-Topic
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<GetTopicAttributesResponse xmlns="https://sns.amazonaws.com/doc/2010-03-31/">
    <GetTopicAttributesResult>
        <Attributes>
            <entry>
                <key>Owner</key>
                <value>123456789012</value>
            </entry>
            <entry>
                <key>Policy</key>
                <value>{&quot;Version&quot;:&quot;2008-10-17&quot;,&quot;Id&quot;:&quot;us-
east-2/698519295917/test__default_policy_ID&quot;,&quot;Statement&quot;:
[{&quot;Effect&quot;:&quot;Allow&quot;,&quot;Sid&quot;:&quot;us-
east-2/698519295917/test__default_statement_ID&quot;,&quot;Principal&quot;:
{&quot;AWS&quot;:&quot;*&quot;},&quot;Action&quot;:
[&quot;SNS:GetTopicAttributes&quot;,&quot;SNS:SetTopicAttributes&quot;,&quot;SNS:AddPermission&quot;,&q
east-2:698519295917:test&quot;,&quot;Condition&quot;:{&quot;StringLike&quot;:
{&quot;AWS:SourceArn&quot;:&quot;arn:aws:*:*:698519295917:*&quot;}}}]}</value>
            </entry>
```

```
        <entry>
            <key>TopicArn</key>
            <value>arn:aws:sns:us-east-2:123456789012:My-Topic</value>
        </entry>
    </Attributes>
</GetTopicAttributesResult>
<ResponseMetadata>
    <RequestId>057f074c-33a7-11df-9540-99d0768312d3</RequestId>
</ResponseMetadata>
</GetTopicAttributesResponse>
```

See Also

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

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

ListEndpointsByPlatformApplication

Lists the endpoints and endpoint attributes for devices in a supported push notification service, such as GCM (Firebase Cloud Messaging) and APNS. The results for `ListEndpointsByPlatformApplication` are paginated and return a limited list of endpoints, up to 100. If additional records are available after the first page results, then a NextToken string will be returned. To receive the next page, you call `ListEndpointsByPlatformApplication` again using the NextToken string received from the previous call. When there are no more records to return, NextToken will be null. For more information, see Using Amazon SNS Mobile Push Notifications.

This action is throttled at 30 transactions per second (TPS).

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

NextToken

NextToken string is used when calling ListEndpointsByPlatformApplication action to retrieve additional records that are available after the first page results.

Type: String

Required: No

PlatformApplicationArn

PlatformApplicationArn for ListEndpointsByPlatformApplicationInput action.

Type: String

Required: Yes

Response Elements

The following elements are returned by the service.

Endpoints.member.N

Endpoints returned for ListEndpointsByPlatformApplication action.

Type: Array of Endpoint (p. 91) objects

NextToken

NextToken string is returned when calling ListEndpointsByPlatformApplication action if additional records are available after the first page results.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

NotFound

Indicates that the requested resource does not exist.

HTTP Status Code: 404

Example

The structure of `AUTHPARAMS` depends on the signature of the API request. For more information, see Examples of Signed Signature Version 4 Requests in the *Amazon Web Services General Reference*.

Example

Sample Request

```
https://sns.us-west-2.amazonaws.com/?Action=ListEndpointsByPlatformApplication
&PlatformApplicationArn=arn%3Aaws%3Asns%3Aus-west-2%3A123456789012%3Aapp%2FGCM%2Fgcmpushapp
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<ListEndpointsByPlatformApplicationResponse xmlns="https://sns.amazonaws.com/
doc/2010-03-31/">
    <ListEndpointsByPlatformApplicationResult>
        <Endpoints>
            <member>
                <EndpointArn>arn:aws:sns:us-west-2:123456789012:endpoint/GCM/
gcmpushapp/5e3e9847-3183-3f18-a7e8-671c3a57d4b3</EndpointArn>
                <Attributes>
                    <entry>
                        <key>Enabled</key>
                        <value>true</value>
                    </entry>
                    <entry>
                        <key>CustomUserData</key>
                        <value>UserId=27576823</value>
                    </entry>
                    <entry>
                        <key>Token</key>

 <value>APA91bGi7fFachkC1xjlqT66VYEucGHochmf1VQAr9k...jsM0PKPxKhddCzx6paEsyay9Zn3D4wNUJb8m6HZrBEXAMPLE<
value>
                    </entry>
                </Attributes>
            </member>
        </Endpoints>
```

```
        </ListEndpointsByPlatformApplicationResult>
        <ResponseMetadata>
            <RequestId>9a48768c-dac8-5a60-aec0-3cc27ea08d96</RequestId>
        </ResponseMetadata>
</ListEndpointsByPlatformApplicationResponse>
```

See Also

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

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

ListPhoneNumbersOptedOut

Returns a list of phone numbers that are opted out, meaning you cannot send SMS messages to them.

The results for `ListPhoneNumbersOptedOut` are paginated, and each page returns up to 100 phone numbers. If additional phone numbers are available after the first page of results, then a `NextToken` string will be returned. To receive the next page, you call `ListPhoneNumbersOptedOut` again using the `NextToken` string received from the previous call. When there are no more records to return, `NextToken` will be null.

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

nextToken

A `NextToken` string is used when you call the `ListPhoneNumbersOptedOut` action to retrieve additional records that are available after the first page of results.

Type: String

Required: No

Response Elements

The following elements are returned by the service.

nextToken

A `NextToken` string is returned when you call the `ListPhoneNumbersOptedOut` action if additional records are available after the first page of results.

Type: String

phoneNumbers.member.N

A list of phone numbers that are opted out of receiving SMS messages. The list is paginated, and each page can contain up to 100 phone numbers.

Type: Array of strings

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

Throttled

Indicates that the rate at which requests have been submitted for this action exceeds the limit for your account.

HTTP Status Code: 429

See Also

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

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

ListPlatformApplications

Lists the platform application objects for the supported push notification services, such as APNS and GCM (Firebase Cloud Messaging). The results for `ListPlatformApplications` are paginated and return a limited list of applications, up to 100. If additional records are available after the first page results, then a NextToken string will be returned. To receive the next page, you call `ListPlatformApplications` using the NextToken string received from the previous call. When there are no more records to return, `NextToken` will be null. For more information, see Using Amazon SNS Mobile Push Notifications.

This action is throttled at 15 transactions per second (TPS).

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

NextToken

NextToken string is used when calling ListPlatformApplications action to retrieve additional records that are available after the first page results.

Type: String

Required: No

Response Elements

The following elements are returned by the service.

NextToken

NextToken string is returned when calling ListPlatformApplications action if additional records are available after the first page results.

Type: String

PlatformApplications.member.N

Platform applications returned when calling ListPlatformApplications action.

Type: Array of PlatformApplication (p. 93) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

Example

The structure of `AUTHPARAMS` depends on the signature of the API request. For more information, see Examples of Signed Signature Version 4 Requests in the *Amazon Web Services General Reference*.

Example

Sample Request

```
https://sns.us-west-2.amazonaws.com/?Action=ListPlatformApplications
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<ListPlatformApplicationsResponse xmlns="https://sns.amazonaws.com/doc/2010-03-31/">
    <ListPlatformApplicationsResult>
        <PlatformApplications>
            <member>
                <PlatformApplicationArn>arn:aws:sns:us-west-2:123456789012:app/
APNS_SANDBOX/apnspushapp</PlatformApplicationArn>
                <Attributes>
                    <entry>
                        <key>AllowEndpointPolicies</key>
                        <value>false</value>
                    </entry>
                </Attributes>
            </member>
            <member>
                <PlatformApplicationArn>arn:aws:sns:us-west-2:123456789012:app/GCM/
gcmpushapp</PlatformApplicationArn>
                <Attributes>
                    <entry>
                        <key>AllowEndpointPolicies</key>
                        <value>false</value>
                    </entry>
                </Attributes>
            </member>
        </PlatformApplications>
    </ListPlatformApplicationsResult>
    <ResponseMetadata>
        <RequestId>315a335e-85d8-52df-9349-791283cbb529</RequestId>
    </ResponseMetadata>
</ListPlatformApplicationsResponse>
```

See Also

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

- AWS Command Line Interface

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

ListSubscriptions

Returns a list of the requester's subscriptions. Each call returns a limited list of subscriptions, up to 100. If there are more subscriptions, a `NextToken` is also returned. Use the `NextToken` parameter in a new `ListSubscriptions` call to get further results.

This action is throttled at 30 transactions per second (TPS).

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

NextToken

Token returned by the previous `ListSubscriptions` request.

Type: String

Required: No

Response Elements

The following elements are returned by the service.

NextToken

Token to pass along to the next `ListSubscriptions` request. This element is returned if there are more subscriptions to retrieve.

Type: String

Subscriptions.member.N

A list of subscriptions.

Type: Array of Subscription (p. 94) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

Example

The structure of AUTHPARAMS depends on the signature of the API request. For more information, see Examples of Signed Signature Version 4 Requests in the *Amazon Web Services General Reference*.

Example

Sample Request

```
https://sns.us-east-2.amazonaws.com/?Action=ListSubscriptions
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<ListSubscriptionsResponse xmlns="https://sns.amazonaws.com/doc/2010-03-31/">
    <ListSubscriptionsResult>
        <Subscriptions>
            <member>
                <TopicArn>arn:aws:sns:us-east-2:698519295917:My-Topic</TopicArn>
                <Protocol>email</Protocol>
                <SubscriptionArn>arn:aws:sns:us-east-2:123456789012:My-
Topic:80289ba6-0fd4-4079-afb4-ce8c8260f0ca</SubscriptionArn>
                <Owner>123456789012</Owner>
                <Endpoint>example@amazon.com</Endpoint>
            </member>
        </Subscriptions>
    </ListSubscriptionsResult>
    <ResponseMetadata>
        <RequestId>384ac68d-3775-11df-8963-01868b7c937a</RequestId>
    </ResponseMetadata>
</ListSubscriptionsResponse>
```

See Also

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

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

ListSubscriptionsByTopic

Returns a list of the subscriptions to a specific topic. Each call returns a limited list of subscriptions, up to 100. If there are more subscriptions, a `NextToken` is also returned. Use the `NextToken` parameter in a new `ListSubscriptionsByTopic` call to get further results.

This action is throttled at 30 transactions per second (TPS).

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

NextToken

Token returned by the previous `ListSubscriptionsByTopic` request.

Type: String

Required: No

TopicArn

The ARN of the topic for which you wish to find subscriptions.

Type: String

Required: Yes

Response Elements

The following elements are returned by the service.

NextToken

Token to pass along to the next `ListSubscriptionsByTopic` request. This element is returned if there are more subscriptions to retrieve.

Type: String

Subscriptions.member.N

A list of subscriptions.

Type: Array of Subscription (p. 94) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

NotFound

Indicates that the requested resource does not exist.

HTTP Status Code: 404

Example

The structure of `AUTHPARAMS` depends on the signature of the API request. For more information, see Examples of Signed Signature Version 4 Requests in the *Amazon Web Services General Reference*.

Example

Sample Request

```
https://sns.us-east-2.amazonaws.com/?Action=ListSubscriptionsByTopic
&TopicArn=arn%3Aaws%3Asns%3Aus-east-2%3A123456789012%3AMy-Topic
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<ListSubscriptionsByTopicResponse xmlns="https://sns.amazonaws.com/doc/2010-03-31/">
    <ListSubscriptionsByTopicResult>
        <Subscriptions>
            <member>
                <TopicArn>arn:aws:sns:us-east-2:123456789012:My-Topic</TopicArn>
                <Protocol>email</Protocol>
                <SubscriptionArn>arn:aws:sns:us-east-2:123456789012:My-
Topic:80289ba6-0fd4-4079-afb4-ce8c8260f0ca</SubscriptionArn>
                <Owner>123456789012</Owner>
                <Endpoint>example@amazon.com</Endpoint>
            </member>
        </Subscriptions>
    </ListSubscriptionsByTopicResult>
    <ResponseMetadata>
        <RequestId>b9275252-3774-11df-9540-99d0768312d3</RequestId>
    </ResponseMetadata>
</ListSubscriptionsByTopicResponse>
```

See Also

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

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

- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

ListTagsForResource

List all tags added to the specified Amazon SNS topic. For an overview, see Amazon SNS Tags in the *Amazon Simple Notification Service Developer Guide*.

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

ResourceArn

The ARN of the topic for which to list tags.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1011.

Required: Yes

Response Elements

The following element is returned by the service.

Tags.member.N

The tags associated with the specified topic.

Type: Array of Tag (p. 95) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

ConcurrentAccess

Can't perform multiple operations on a tag simultaneously. Perform the operations sequentially.

HTTP Status Code: 400

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

ResourceNotFound

Can't tag resource. Verify that the topic exists.

HTTP Status Code: 404

TagPolicy

The request doesn't comply with the IAM tag policy. Correct your request and then retry it.

HTTP Status Code: 400

Example

The structure of `AUTHPARAMS` depends on the signature of the API request. For more information, see Examples of Signed Signature Version 4 Requests in the *Amazon Web Services General Reference*.

Example

Sample Request

```
http://sns.us-west-2.amazonaws.com/&Action=ListTagsForResource
&ResourceArn=arn%3Aaws%3Asns%3Aus-west-2%3A123456789012%3Atagging
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<ListTagsForResourceResponse>
    <ListTagsForResourceResult>
        <Tags>
            <member>
                <Value>tagValue</Value>
                <Key>tagKey</Key>
            </member>
        </Tags>
    </ListTagsForResourceResult>
    <ResponseMetadata>
        <RequestId>123456a7-b8c9-01de-fg2h-3456i7j890k1</RequestId>
    </ResponseMetadata>
</ListTagsForResourceResponse>
```

See Also

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

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

ListTopics

Returns a list of the requester's topics. Each call returns a limited list of topics, up to 100. If there are more topics, a `NextToken` is also returned. Use the `NextToken` parameter in a new `ListTopics` call to get further results.

This action is throttled at 30 transactions per second (TPS).

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

NextToken

Token returned by the previous `ListTopics` request.

Type: String

Required: No

Response Elements

The following elements are returned by the service.

NextToken

Token to pass along to the next `ListTopics` request. This element is returned if there are additional topics to retrieve.

Type: String

Topics.member.N

A list of topic ARNs.

Type: Array of Topic (p. 96) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

See Also

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

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

OptInPhoneNumber

Use this request to opt in a phone number that is opted out, which enables you to resume sending SMS messages to the number.

You can opt in a phone number only once every 30 days.

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

phoneNumber

　　The phone number to opt in.

　　Type: String

　　Required: Yes

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

　　Indicates that the user has been denied access to the requested resource.

　　HTTP Status Code: 403

InternalError

　　Indicates an internal service error.

　　HTTP Status Code: 500

InvalidParameter

　　Indicates that a request parameter does not comply with the associated constraints.

　　HTTP Status Code: 400

Throttled

　　Indicates that the rate at which requests have been submitted for this action exceeds the limit for your account.

　　HTTP Status Code: 429

See Also

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

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

- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

Publish

Sends a message to an Amazon SNS topic, a text message (SMS message) directly to a phone number, or a message to a mobile platform endpoint (when you specify the `TargetArn`).

If you send a message to a topic, Amazon SNS delivers the message to each endpoint that is subscribed to the topic. The format of the message depends on the notification protocol for each subscribed endpoint.

When a `messageId` is returned, the message has been saved and Amazon SNS will attempt to deliver it shortly.

To use the `Publish` action for sending a message to a mobile endpoint, such as an app on a Kindle device or mobile phone, you must specify the EndpointArn for the TargetArn parameter. The EndpointArn is returned when making a call with the `CreatePlatformEndpoint` action. The second example below shows a request and response for publishing to a mobile endpoint.

For more information about formatting messages, see Send Custom Platform-Specific Payloads in Messages to Mobile Devices.

> **Important**
> You can publish messages only to topics and endpoints in the same AWS Region.

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

Message

The message you want to send.

If you are publishing to a topic and you want to send the same message to all transport protocols, include the text of the message as a String value. If you want to send different messages for each transport protocol, set the value of the `MessageStructure` parameter to `json` and use a JSON object for the `Message` parameter. See the Examples section for the format of the JSON object.

Constraints:
- With the exception of SMS, messages must be UTF-8 encoded strings and at most 256 KB in size (262,144 bytes, not 262,144 characters).
- For SMS, each message can contain up to 140 characters. This character limit depends on the encoding schema. For example, an SMS message can contain 160 GSM characters, 140 ASCII characters, or 70 UCS-2 characters.

  If you publish a message that exceeds this size limit, Amazon SNS sends the message as multiple messages, each fitting within the size limit. Messages aren't truncated mid-word but are cut off at whole-word boundaries.

  The total size limit for a single SMS `Publish` action is 1,600 characters.

JSON-specific constraints:
- Keys in the JSON object that correspond to supported transport protocols must have simple JSON string values.
- The values will be parsed (unescaped) before they are used in outgoing messages.
- Outbound notifications are JSON encoded (meaning that the characters will be reescaped for sending).

- Values have a minimum length of 0 (the empty string, "", is allowed).
- Values have a maximum length bounded by the overall message size (so, including multiple protocols may limit message sizes).
- Non-string values will cause the key to be ignored.
- Keys that do not correspond to supported transport protocols are ignored.
- Duplicate keys are not allowed.
- Failure to parse or validate any key or value in the message will cause the `Publish` call to return an error (no partial delivery).

Type: String

Required: Yes

MessageAttributes , MessageAttributes.entry.N.Name (key), MessageAttributes.entry.N.Value (value)

Message attributes for Publish action.

Type: String to [MessageAttributeValue (p. 92)](#) object map

Required: No

MessageStructure

Set `MessageStructure` to `json` if you want to send a different message for each protocol. For example, using one publish action, you can send a short message to your SMS subscribers and a longer message to your email subscribers. If you set `MessageStructure` to `json`, the value of the `Message` parameter must:

- be a syntactically valid JSON object; and
- contain at least a top-level JSON key of "default" with a value that is a string.

You can define other top-level keys that define the message you want to send to a specific transport protocol (e.g., "http").

Valid value: `json`

Type: String

Required: No

PhoneNumber

The phone number to which you want to deliver an SMS message. Use E.164 format.

If you don't specify a value for the `PhoneNumber` parameter, you must specify a value for the `TargetArn` or `TopicArn` parameters.

Type: String

Required: No

Subject

Optional parameter to be used as the "Subject" line when the message is delivered to email endpoints. This field will also be included, if present, in the standard JSON messages delivered to other endpoints.

Constraints: Subjects must be ASCII text that begins with a letter, number, or punctuation mark; must not include line breaks or control characters; and must be less than 100 characters long.

Type: String

Required: No

TargetArn

If you don't specify a value for the `TargetArn` parameter, you must specify a value for the `PhoneNumber` or `TopicArn` parameters.

Type: String

Required: No

TopicArn

The topic you want to publish to.

If you don't specify a value for the `TopicArn` parameter, you must specify a value for the `PhoneNumber` or `TargetArn` parameters.

Type: String

Required: No

Response Elements

The following element is returned by the service.

MessageId

Unique identifier assigned to the published message.

Length Constraint: Maximum 100 characters

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

EndpointDisabled

Exception error indicating endpoint disabled.

HTTP Status Code: 400

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

InvalidSecurity

The credential signature isn't valid. You must use an HTTPS endpoint and sign your request using Signature Version 4.

HTTP Status Code: 403

KMSAccessDenied

The ciphertext references a key that doesn't exist or that you don't have access to.

HTTP Status Code: 400

KMSDisabled

The request was rejected because the specified customer master key (CMK) isn't enabled.

HTTP Status Code: 400

KMSInvalidState

The request was rejected because the state of the specified resource isn't valid for this request. For more information, see How Key State Affects Use of a Customer Master Key in the *AWS Key Management Service Developer Guide*.

HTTP Status Code: 400

KMSNotFound

The request was rejected because the specified entity or resource can't be found.

HTTP Status Code: 400

KMSOptInRequired

The AWS access key ID needs a subscription for the service.

HTTP Status Code: 403

KMSThrottling

The request was denied due to request throttling. For more information about throttling, see Limits in the *AWS Key Management Service Developer Guide.*

HTTP Status Code: 400

NotFound

Indicates that the requested resource does not exist.

HTTP Status Code: 404

ParameterValueInvalid

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

PlatformApplicationDisabled

Exception error indicating platform application disabled.

HTTP Status Code: 400

Examples

The structure of AUTHPARAMS depends on the signature of the API request. For more information, see Examples of Signed Signature Version 4 Requests in the *Amazon Web Services General Reference*.

Example

Sample Request

```
https://sns.us-west-2.amazonaws.com/?Action=Publish
&TargetArn=arn%3Aaws%3Asns%3Aus-west-2%3A803981987763%3Aendpoint%2FAPNS_SANDBOX%2Fpushapp
%2F98e9ced9-f136-3893-9d60-776547eafebb
&Message=%7B%22default%22%3A%22This+is+the+default+Message%22%2C%22APNS_SANDBOX%22%3A
%22%7B+%5C%22aps%5C%22+%3A+%7B+%5C%22alert%5C%22+%3A+%5C%22You+have+got+email.%5C%22%2C+%5C
%22badge%5C%22+%3A+9%2C%5C%22sound%5C%22+%3A%5C%22default%5C%22%7D%7D%22%7D
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<PublishResponse xmlns="https://sns.amazonaws.com/doc/2010-03-31/">
    <PublishResult>
        <MessageId>567910cd-659e-55d4-8ccb-5aaf14679dc0</MessageId>
    </PublishResult>
    <ResponseMetadata>
        <RequestId>d74b8436-ae13-5ab4-a9ff-ce54dfea72a0</RequestId>
    </ResponseMetadata>
</PublishResponse>
```

Example

The following example publishes a different messages to each protocol:

The JSON format for `Message` is as follows:

```
{
    "default": "A message.",
    "email": "A message for email.",
    "email - json ": "A message for email (JSON).",
    "http": "A message for HTTP.",
    "https": "A message for HTTPS.",
    "sqs": "A message for Amazon SQS."
}
```

Sample Request

```
https://sns.us-east-2.amazonaws.com/?Action=Publish
&TopicArn=arn%3Aaws%3Asns%3Aus-east-2%3A698519295917%3AMy-Topic
&Subject=My%20first%20message
&Message=Hello%20world%21
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<PublishResponse xmlns="https://sns.amazonaws.com/doc/2010-03-31/">
    <PublishResult>
        <MessageId>94f20ce6-13c5-43a0-9a9e-ca52d816e90b</MessageId>
    </PublishResult>
    <ResponseMetadata>
        <RequestId>f187a3c1-376f-11df-8963-01868b7c937a</RequestId>
    </ResponseMetadata>
```

```
</PublishResponse>
```

See Also

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

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

RemovePermission

Removes a statement from a topic's access control policy.

Request Parameters

For information about the parameters that are common to all actions, see Common
Parameters (p. 97).

Label

The unique label of the statement you want to remove.

Type: String

Required: Yes

TopicArn

The ARN of the topic whose access control policy you wish to modify.

Type: String

Required: Yes

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

NotFound

Indicates that the requested resource does not exist.

HTTP Status Code: 404

Example

The structure of `AUTHPARAMS` depends on the signature of the API request. For more information, see
Examples of Signed Signature Version 4 Requests in the *Amazon Web Services General Reference*.

Example

Sample Request

```
https://sns.us-east-2.amazonaws.com/?Action=RemovePermission
&TopicArn=arn%3Aaws%3Asns%3Aus-east-2%3A123456789012%3AMy-Test
&Label=NewPermission
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<RemovePermissionResponse xmlns="https://sns.amazonaws.com/doc/2010-03-31/">
    <ResponseMetadata>
        <RequestId>d170b150-33a8-11df-995a-2d6fbe836cc1</RequestId>
    </ResponseMetadata>
</RemovePermissionResponse>
```

See Also

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

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

SetEndpointAttributes

Sets the attributes for an endpoint for a device on one of the supported push notification services, such as GCM (Firebase Cloud Messaging) and APNS. For more information, see Using Amazon SNS Mobile Push Notifications.

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

Attributes , Attributes.entry.N.key (key), Attributes.entry.N.value (value)

A map of the endpoint attributes. Attributes in this map include the following:

- `CustomUserData` – arbitrary user data to associate with the endpoint. Amazon SNS does not use this data. The data must be in UTF-8 format and less than 2KB.
- `Enabled` – flag that enables/disables delivery to the endpoint. Amazon SNS will set this to false when a notification service indicates to Amazon SNS that the endpoint is invalid. Users can set it back to true, typically after updating Token.
- `Token` – device token, also referred to as a registration id, for an app and mobile device. This is returned from the notification service when an app and mobile device are registered with the notification service.

Type: String to string map

Required: Yes

EndpointArn

EndpointArn used for SetEndpointAttributes action.

Type: String

Required: Yes

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

NotFound

Indicates that the requested resource does not exist.

HTTP Status Code: 404

Example

The structure of `AUTHPARAMS` depends on the signature of the API request. For more information, see Examples of Signed Signature Version 4 Requests in the *Amazon Web Services General Reference*.

Example

Sample Request

```
https://sns.us-west-2.amazonaws.com/?Action=SetEndpointAttributes
&EndpointArn=arn%3Aaws%3Asns%3Aus-west-2%3A123456789012%3Aendpoint%2FGCM%2Fgcmpushapp
%2F5e3e9847-3183-3f18-a7e8-671c3a57d4b3
&Attributes.entry.1.key=CustomUserData
&Attributes.entry.1.value=My+custom+userdata
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<SetEndpointAttributesResponse xmlns="https://sns.amazonaws.com/doc/2010-03-31/">
    <ResponseMetadata>
        <RequestId>2fe0bfc7-3e85-5ee5-a9e2-f58b35e85f6a</RequestId>
    </ResponseMetadata>
</SetEndpointAttributesResponse>
```

See Also

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

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

SetPlatformApplicationAttributes

Sets the attributes of the platform application object for the supported push notification services, such as APNS and GCM (Firebase Cloud Messaging). For more information, see Using Amazon SNS Mobile Push Notifications. For information on configuring attributes for message delivery status, see Using Amazon SNS Application Attributes for Message Delivery Status.

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

Attributes , Attributes.entry.N.key (key), Attributes.entry.N.value (value)

A map of the platform application attributes. Attributes in this map include the following:

- `PlatformCredential` – The credential received from the notification service. For `APNS` and `APNS_SANDBOX`, `PlatformCredential` is `private key`. For `GCM` (Firebase Cloud Messaging), `PlatformCredential` is `API key`. For `ADM`, `PlatformCredential` is `client secret`.
- `PlatformPrincipal` – The principal received from the notification service. For `APNS` and `APNS_SANDBOX`, `PlatformPrincipal` is `SSL certificate`. For `GCM` (Firebase Cloud Messaging), there is no `PlatformPrincipal`. For `ADM`, `PlatformPrincipal` is `client id`.
- `EventEndpointCreated` – Topic ARN to which `EndpointCreated` event notifications are sent.
- `EventEndpointDeleted` – Topic ARN to which `EndpointDeleted` event notifications are sent.
- `EventEndpointUpdated` – Topic ARN to which `EndpointUpdate` event notifications are sent.
- `EventDeliveryFailure` – Topic ARN to which `DeliveryFailure` event notifications are sent upon Direct Publish delivery failure (permanent) to one of the application's endpoints.
- `SuccessFeedbackRoleArn` – IAM role ARN used to give Amazon SNS write access to use CloudWatch Logs on your behalf.
- `FailureFeedbackRoleArn` – IAM role ARN used to give Amazon SNS write access to use CloudWatch Logs on your behalf.
- `SuccessFeedbackSampleRate` – Sample rate percentage (0-100) of successfully delivered messages.

Type: String to string map

Required: Yes

PlatformApplicationArn

PlatformApplicationArn for SetPlatformApplicationAttributes action.

Type: String

Required: Yes

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

NotFound

Indicates that the requested resource does not exist.

HTTP Status Code: 404

Example

The structure of `AUTHPARAMS` depends on the signature of the API request. For more information, see Examples of Signed Signature Version 4 Requests in the *Amazon Web Services General Reference*.

Example

Sample Request

```
https://sns.us-west-2.amazonaws.com/?Action=SetPlatformApplicationAttributes
&Attributes.entry.1.key=EventEndpointCreated&PlatformApplicationArn=arn%3Aaws%3Asns%3Aus-
west-2%3A123456789012%3Aapp%2FGCM%2Fgcmpushapp
&Attributes.entry.1.value=arn%3Aaws%3Asns%3Aus-west-2%3A123456789012%3Atopicarn
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<SetPlatformApplicationAttributesResponse xmlns="https://sns.amazonaws.com/
doc/2010-03-31/">
    <ResponseMetadata>
        <RequestId>cf577bcc-b3dc-5463-88f1-3180b9412395</RequestId>
    </ResponseMetadata>
</SetPlatformApplicationAttributesResponse>
```

See Also

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

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

- [AWS SDK for Ruby V2](#)

SetSMSAttributes

Use this request to set the default settings for sending SMS messages and receiving daily SMS usage reports.

You can override some of these settings for a single message when you use the `Publish` action with the `MessageAttributes.entry.N` parameter. For more information, see Sending an SMS Message in the *Amazon SNS Developer Guide*.

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

attributes , attributes.entry.N.key (key), attributes.entry.N.value (value)

> The default settings for sending SMS messages from your account. You can set values for the following attribute names:
>
> `MonthlySpendLimit` – The maximum amount in USD that you are willing to spend each month to send SMS messages. When Amazon SNS determines that sending an SMS message would incur a cost that exceeds this limit, it stops sending SMS messages within minutes.
>
> > **Important**
> > Amazon SNS stops sending SMS messages within minutes of the limit being crossed. During that interval, if you continue to send SMS messages, you will incur costs that exceed your limit.
>
> By default, the spend limit is set to the maximum allowed by Amazon SNS. If you want to raise the limit, submit an SNS Limit Increase case. For **New limit value**, enter your desired monthly spend limit. In the **Use Case Description** field, explain that you are requesting an SMS monthly spend limit increase.
>
> `DeliveryStatusIAMRole` – The ARN of the IAM role that allows Amazon SNS to write logs about SMS deliveries in CloudWatch Logs. For each SMS message that you send, Amazon SNS writes a log that includes the message price, the success or failure status, the reason for failure (if the message failed), the message dwell time, and other information.
>
> `DeliveryStatusSuccessSamplingRate` – The percentage of successful SMS deliveries for which Amazon SNS will write logs in CloudWatch Logs. The value can be an integer from 0 - 100. For example, to write logs only for failed deliveries, set this value to `0`. To write logs for 10% of your successful deliveries, set it to `10`.
>
> `DefaultSenderID` – A string, such as your business brand, that is displayed as the sender on the receiving device. Support for sender IDs varies by country. The sender ID can be 1 - 11 alphanumeric characters, and it must contain at least one letter.
>
> `DefaultSMSType` – The type of SMS message that you will send by default. You can assign the following values:
>
> - `Promotional` – (Default) Noncritical messages, such as marketing messages. Amazon SNS optimizes the message delivery to incur the lowest cost.
> - `Transactional` – Critical messages that support customer transactions, such as one-time passcodes for multi-factor authentication. Amazon SNS optimizes the message delivery to achieve the highest reliability.
>
> `UsageReportS3Bucket` – The name of the Amazon S3 bucket to receive daily SMS usage reports from Amazon SNS. Each day, Amazon SNS will deliver a usage report as a CSV file to the bucket. The

report includes the following information for each SMS message that was successfully delivered by your account:

- Time that the message was published (in UTC)
- Message ID
- Destination phone number
- Message type
- Delivery status
- Message price (in USD)
- Part number (a message is split into multiple parts if it is too long for a single message)
- Total number of parts

To receive the report, the bucket must have a policy that allows the Amazon SNS service principle to perform the `s3:PutObject` and `s3:GetBucketLocation` actions.

For an example bucket policy and usage report, see Monitoring SMS Activity in the *Amazon SNS Developer Guide*.

Type: String to string map

Required: Yes

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

Throttled

Indicates that the rate at which requests have been submitted for this action exceeds the limit for your account.

HTTP Status Code: 429

See Also

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

- AWS Command Line Interface
- AWS SDK for .NET

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

SetSubscriptionAttributes

Allows a subscription owner to set an attribute of the subscription to a new value.

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

AttributeName

A map of attributes with their corresponding values.

The following lists the names, descriptions, and values of the special request parameters that the `SetTopicAttributes` action uses:

- `DeliveryPolicy` – The policy that defines how Amazon SNS retries failed deliveries to HTTP/S endpoints.
- `FilterPolicy` – The simple JSON object that lets your subscriber receive only a subset of messages, rather than receiving every message published to the topic.
- `RawMessageDelivery` – When set to `true`, enables raw message delivery to Amazon SQS or HTTP/S endpoints. This eliminates the need for the endpoints to process JSON formatting, which is otherwise created for Amazon SNS metadata.
- `RedrivePolicy` – When specified, sends undeliverable messages to the specified Amazon SQS dead-letter queue. Messages that can't be delivered due to client errors (for example, when the subscribed endpoint is unreachable) or server errors (for example, when the service that powers the subscribed endpoint becomes unavailable) are held in the dead-letter queue for further analysis or reprocessing.

Type: String

Required: Yes

AttributeValue

The new value for the attribute in JSON format.

Type: String

Required: No

SubscriptionArn

The ARN of the subscription to modify.

Type: String

Required: Yes

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

FilterPolicyLimitExceeded

Indicates that the number of filter polices in your AWS account exceeds the limit. To add more filter polices, submit an SNS Limit Increase case in the AWS Support Center.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

NotFound

Indicates that the requested resource does not exist.

HTTP Status Code: 404

Example

The structure of `AUTHPARAMS` depends on the signature of the API request. For more information, see Examples of Signed Signature Version 4 Requests in the *Amazon Web Services General Reference*.

Example

The following example sets the delivery policy to 5 retries.

The JSON format for `AttributeValue` is as follows:

```
{
    "healthyRetryPolicy": {
        "minDelayTarget": <int>,
        "maxDelayTarget": <int>,
        "numRetries": <int>,
        "numMaxDelayRetries": <int>,
        "backoffFunction": "<linear|arithmetic|geometric|exponential>"
    },
    "throttlePolicy": {
        "maxReceivesPerSecond": <int>
    }
}
```

Sample Request

```
https://sns.us-east-2.amazonaws.com/?Action=SetSubscriptionAttributes
&SubscriptionArn=arn%3Aaws%3Asns%3Aus-east-2%3A123456789012%3AMy-Topic
%3A80289ba6-0fd4-4079-afb4-ce8c8260f0ca
&AttributeName=DeliveryPolicy
&AttributeValue={"healthyRetryPolicy":{"numRetries":5}}
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<SetSubscriptionAttributesResponse xmlns="https://sns.amazonaws.com/doc/2010-03-31/">
    <ResponseMetadata>
        <RequestId>a8763b99-33a7-11df-a9b7-05d48da6f042</RequestId>
    </ResponseMetadata>
</SetSubscriptionAttributesResponse>
```

See Also

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

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

SetTopicAttributes

Allows a topic owner to set an attribute of the topic to a new value.

Request Parameters

For information about the parameters that are common to all actions, see Common
Parameters (p. 97).

AttributeName

A map of attributes with their corresponding values.

The following lists the names, descriptions, and values of the special request parameters that the
`SetTopicAttributes` action uses:

- `DeliveryPolicy` – The policy that defines how Amazon SNS retries failed deliveries to HTTP/S
  endpoints.
- `DisplayName` – The display name to use for a topic with SMS subscriptions.
- `Policy` – The policy that defines who can access your topic. By default, only the topic owner can
  publish or subscribe to the topic.

The following attribute applies only to server-side-encryption:

- `KmsMasterKeyId` - The ID of an AWS-managed customer master key (CMK) for Amazon SNS or a
  custom CMK. For more information, see Key Terms. For more examples, see KeyId in the *AWS Key
  Management Service API Reference*.

Type: String

Required: Yes

AttributeValue

The new value for the attribute.

Type: String

Required: No

TopicArn

The ARN of the topic to modify.

Type: String

Required: Yes

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

InvalidSecurity

The credential signature isn't valid. You must use an HTTPS endpoint and sign your request using Signature Version 4.

HTTP Status Code: 403

NotFound

Indicates that the requested resource does not exist.

HTTP Status Code: 404

Examples

The structure of `AUTHPARAMS` depends on the signature of the API request. For more information, see Examples of Signed Signature Version 4 Requests in the *Amazon Web Services General Reference*.

Example

The following example sets the `DisplayName` attribute to `MyTopicName`.

Sample Request

```
https://sns.us-east-2.amazonaws.com/?Action=SetTopicAttributes
&TopicArn=arn%3Aaws%3Asns%3Aus-east-2%3A123456789012%3AMy-Topic
&AttributeName=DisplayName
&AttributeValue=MyTopicName
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<SetTopicAttributesResponse xmlns="https://sns.amazonaws.com/doc/2010-03-31/">
    <ResponseMetadata>
        <RequestId>a8763b99-33a7-11df-a9b7-05d48da6f042</RequestId>
    </ResponseMetadata>
</SetTopicAttributesResponse>
```

Example

The following example sets the delivery policy to 5 retries.

The JSON format for `AttributeValue` is as follows:

```
"http": {
    "defaultHealthyRetryPolicy": {
        "minDelayTarget": <int>,
        "maxDelayTarget": <int>,
        "numRetries": <int>,
```

```
        "numMaxDelayRetries": <int>,
        "backoffFunction": "<linear|arithmetic|geometric|exponential>"
    },
    "disableSubscriptionOverrides": <boolean>,
    "defaultThrottlePolicy": {
        "maxReceivesPerSecond": <int>
    }
}
```

Sample Request

```
https://sns.us-east-2.amazonaws.com/?Action=SetTopicAttributes
&TopicArn=arn%3Aaws%3Asns%3Aus-east-2%3A123456789012%3AMy-Topic
&AttributeName=DeliveryPolicy
&AttributeValue={"http":{"defaultHealthyRetryPolicy":{"numRetries":5}}}
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<SetTopicAttributesResponse xmlns="https://sns.amazonaws.com/doc/2010-03-31/">
    <ResponseMetadata>
        <RequestId>a8763b99-33a7-11df-a9b7-05d48da6f042</RequestId>
    </ResponseMetadata>
</SetTopicAttributesResponse>
```

See Also

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

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

Subscribe

Prepares to subscribe an endpoint by sending the endpoint a confirmation message. To actually create a subscription, the endpoint owner must call the `ConfirmSubscription` action with the token from the confirmation message. Confirmation tokens are valid for three days.

This action is throttled at 100 transactions per second (TPS).

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

Attributes , Attributes.entry.N.key (key), Attributes.entry.N.value (value)

A map of attributes with their corresponding values.

The following lists the names, descriptions, and values of the special request parameters that the `SetTopicAttributes` action uses:

- `DeliveryPolicy` – The policy that defines how Amazon SNS retries failed deliveries to HTTP/S endpoints.
- `FilterPolicy` – The simple JSON object that lets your subscriber receive only a subset of messages, rather than receiving every message published to the topic.
- `RawMessageDelivery` – When set to `true`, enables raw message delivery to Amazon SQS or HTTP/S endpoints. This eliminates the need for the endpoints to process JSON formatting, which is otherwise created for Amazon SNS metadata.
- `RedrivePolicy` – When specified, sends undeliverable messages to the specified Amazon SQS dead-letter queue. Messages that can't be delivered due to client errors (for example, when the subscribed endpoint is unreachable) or server errors (for example, when the service that powers the subscribed endpoint becomes unavailable) are held in the dead-letter queue for further analysis or reprocessing.

Type: String to string map

Required: No

Endpoint

The endpoint that you want to receive notifications. Endpoints vary by protocol:

- For the `http` protocol, the endpoint is an URL beginning with `http://`
- For the `https` protocol, the endpoint is a URL beginning with `https://`
- For the `email` protocol, the endpoint is an email address
- For the `email-json` protocol, the endpoint is an email address
- For the `sms` protocol, the endpoint is a phone number of an SMS-enabled device
- For the `sqs` protocol, the endpoint is the ARN of an Amazon SQS queue
- For the `application` protocol, the endpoint is the EndpointArn of a mobile app and device.
- For the `lambda` protocol, the endpoint is the ARN of an Amazon Lambda function.

Type: String

Required: No

Protocol

The protocol you want to use. Supported protocols include:

- `http` – delivery of JSON-encoded message via HTTP POST
- `https` – delivery of JSON-encoded message via HTTPS POST
- `email` – delivery of message via SMTP
- `email-json` – delivery of JSON-encoded message via SMTP
- `sms` – delivery of message via SMS
- `sqs` – delivery of JSON-encoded message to an Amazon SQS queue
- `application` – delivery of JSON-encoded message to an EndpointArn for a mobile app and device.
- `lambda` – delivery of JSON-encoded message to an Amazon Lambda function.

Type: String

Required: Yes

ReturnSubscriptionArn

Sets whether the response from the `Subscribe` request includes the subscription ARN, even if the subscription is not yet confirmed.

- If you have the subscription ARN returned, the response includes the ARN in all cases, even if the subscription is not yet confirmed.
- If you don't have the subscription ARN returned, in addition to the ARN for confirmed subscriptions, the response also includes the `pending subscription` ARN value for subscriptions that aren't yet confirmed. A subscription becomes confirmed when the subscriber calls the `ConfirmSubscription` action with a confirmation token.

If you set this parameter to `true`, .

The default value is `false`.

Type: Boolean

Required: No

TopicArn

The ARN of the topic you want to subscribe to.

Type: String

Required: Yes

Response Elements

The following element is returned by the service.

SubscriptionArn

The ARN of the subscription if it is confirmed, or the string "pending confirmation" if the subscription requires confirmation. However, if the API request parameter `ReturnSubscriptionArn` is true, then the value is always the subscription ARN, even if the subscription requires confirmation.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

FilterPolicyLimitExceeded

Indicates that the number of filter polices in your AWS account exceeds the limit. To add more filter polices, submit an SNS Limit Increase case in the AWS Support Center.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

InvalidSecurity

The credential signature isn't valid. You must use an HTTPS endpoint and sign your request using Signature Version 4.

HTTP Status Code: 403

NotFound

Indicates that the requested resource does not exist.

HTTP Status Code: 404

SubscriptionLimitExceeded

Indicates that the customer already owns the maximum allowed number of subscriptions.

HTTP Status Code: 403

Example

The structure of `AUTHPARAMS` depends on the signature of the API request. For more information, see Examples of Signed Signature Version 4 Requests in the *Amazon Web Services General Reference*.

Example

The following example Query request subscribes an SQS queue to an SNS topic. For more information, see Subscribe Queue to Amazon SNS Topic in the *Amazon SQS Developer Guide*.

Sample Request

```
https://sns.us-west-2.amazonaws.com/?Action=Subscribe
&TopicArn=arn%3Aaws%3Asns%3Aus-west-2%3A123456789012%3AMyTopic
&Endpoint=arn%3Aaws%3Asqs%3Aus-west-2%3A123456789012%3AMyQueue
&Protocol=sqs
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<SubscribeResponse xmlns="https://sns.amazonaws.com/doc/2010-03-31/">
    <SubscribeResult>
        <SubscriptionArn>arn:aws:sns:us-
west-2:123456789012:MyTopic:6b0e71bd-7e97-4d97-80ce-4a0994e55286</SubscriptionArn>
    </SubscribeResult>
    <ResponseMetadata>
        <RequestId>c4407779-24a4-56fa-982c-3d927f93a775</RequestId>
    </ResponseMetadata>
</SubscribeResponse>
```

See Also

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

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

TagResource

Add tags to the specified Amazon SNS topic. For an overview, see Amazon SNS Tags in the *Amazon SNS Developer Guide*.

When you use topic tags, keep the following guidelines in mind:

- Adding more than 50 tags to a topic isn't recommended.
- Tags don't have any semantic meaning. Amazon SNS interprets tags as character strings.
- Tags are case-sensitive.
- A new tag with a key identical to that of an existing tag overwrites the existing tag.
- Tagging actions are limited to 10 TPS per AWS account, per AWS region. If your application requires a higher throughput, file a technical support request.

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

ResourceArn

The ARN of the topic to which to add tags.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1011.

Required: Yes

Tags.member.N

The tags to be added to the specified topic. A tag consists of a required key and an optional value.

Type: Array of Tag (p. 95) objects

Required: Yes

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

ConcurrentAccess

Can't perform multiple operations on a tag simultaneously. Perform the operations sequentially.

HTTP Status Code: 400

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

ResourceNotFound

Can't tag resource. Verify that the topic exists.

HTTP Status Code: 404

StaleTag

A tag has been added to a resource with the same ARN as a deleted resource. Wait a short while and then retry the operation.

HTTP Status Code: 400

TagLimitExceeded

Can't add more than 50 tags to a topic.

HTTP Status Code: 400

TagPolicy

The request doesn't comply with the IAM tag policy. Correct your request and then retry it.

HTTP Status Code: 400

Example

The structure of `AUTHPARAMS` depends on the signature of the API request. For more information, see Examples of Signed Signature Version 4 Requests in the *Amazon Web Services General Reference*.

Example

Sample Request

```
http://sns.us-west-2.amazonaws.com/?&Action=TagResource
&ResourceArn=arn%3Aaws%3Asns%3Aus-west-2%3A123456789012%3Atagging
&Tags.Tag.1.Key=tagKey
&Tags.Tag.1.Value=tagValue
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<TagResourceResponse>
    <TagResourceResult/>
    <ResponseMetadata>
        <RequestId>a1b2c3de-fg45-67h8-i9j0-k1lm234n5678</RequestId>
    </ResponseMetadata>
</TagResourceResponse>
```

See Also

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

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++

- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

Unsubscribe

Deletes a subscription. If the subscription requires authentication for deletion, only the owner of the subscription or the topic's owner can unsubscribe, and an AWS signature is required. If the `Unsubscribe` call does not require authentication and the requester is not the subscription owner, a final cancellation message is delivered to the endpoint, so that the endpoint owner can easily resubscribe to the topic if the `Unsubscribe` request was unintended.

This action is throttled at 100 transactions per second (TPS).

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

SubscriptionArn

The ARN of the subscription to be deleted.

Type: String

Required: Yes

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

InternalError

Indicates an internal service error.

HTTP Status Code: 500

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

InvalidSecurity

The credential signature isn't valid. You must use an HTTPS endpoint and sign your request using Signature Version 4.

HTTP Status Code: 403

NotFound

Indicates that the requested resource does not exist.

HTTP Status Code: 404

Example

The structure of `AUTHPARAMS` depends on the signature of the API request. For more information, see Examples of Signed Signature Version 4 Requests in the *Amazon Web Services General Reference*.

Example

Sample Request

```
https://sns.us-east-2.amazonaws.com/?Action=Unsubscribe
&SubscriptionArn=arn%3Aaws%3Asns%3Aus-east-2%3A123456789012%3AMy-Topic
%3A80289ba6-0fd4-4079-afb4-ce8c8260f0ca
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<UnsubscribeResponse xmlns="https://sns.amazonaws.com/doc/2010-03-31/">
    <ResponseMetadata>
        <RequestId>18e0ac39-3776-11df-84c0-b93cc1666b84</RequestId>
    </ResponseMetadata>
</UnsubscribeResponse>
```

See Also

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

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

UntagResource

Remove tags from the specified Amazon SNS topic. For an overview, see Amazon SNS Tags in the *Amazon SNS Developer Guide*.

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 97).

ResourceArn

The ARN of the topic from which to remove tags.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1011.

Required: Yes

TagKeys.member.N

The list of tag keys to remove from the specified topic.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 128.

Required: Yes

Errors

For information about the errors that are common to all actions, see Common Errors (p. 99).

AuthorizationError

Indicates that the user has been denied access to the requested resource.

HTTP Status Code: 403

ConcurrentAccess

Can't perform multiple operations on a tag simultaneously. Perform the operations sequentially.

HTTP Status Code: 400

InvalidParameter

Indicates that a request parameter does not comply with the associated constraints.

HTTP Status Code: 400

ResourceNotFound

Can't tag resource. Verify that the topic exists.

HTTP Status Code: 404

StaleTag

A tag has been added to a resource with the same ARN as a deleted resource. Wait a short while and then retry the operation.

HTTP Status Code: 400

TagLimitExceeded

Can't add more than 50 tags to a topic.

HTTP Status Code: 400

TagPolicy

The request doesn't comply with the IAM tag policy. Correct your request and then retry it.

HTTP Status Code: 400

Example

The structure of `AUTHPARAMS` depends on the signature of the API request. For more information, see Examples of Signed Signature Version 4 Requests in the *Amazon Web Services General Reference*.

Example

Sample Request

```
http://sns.us-east-2.amazonaws.com/?Action=UntagResource
&ResourceArn=arn%3Aaws%3Asns%3Aus-east-1%3A123456789012%3Atagging
&TagKeys.TagKey.1=tagKey
&Version=2010-03-31
&AUTHPARAMS
```

Sample Response

```
<UntagResourceResponse>
    <UntagResourceResult/>
    <ResponseMetadata>
        <RequestId>1a34f567-8bc9-01de-f234-g567h8908i12</RequestId>
    </ResponseMetadata>
</UntagResourceResponse>
```

See Also

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

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

Data Types

The Amazon Simple Notification Service API contains several data types that various actions use. This section describes each data type in detail.

> **Note**
> The order of each element in a data type structure is not guaranteed. Applications should not assume a particular order.

The following data types are supported:

Endpoint

Endpoint for mobile app and device.

Contents

Attributes , Attributes.entry.N.key (key), Attributes.entry.N.value (value)

Attributes for endpoint.

Type: String to string map

Required: No
EndpointArn

EndpointArn for mobile app and device.

Type: String

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2

MessageAttributeValue

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 Publish.

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 currently 256 KB (262,144 bytes). For more information, see Using Amazon SNS Message Attributes.

Contents

BinaryValue

Binary type attributes can store any binary data, for example, compressed data, encrypted data, or images.

Type: Base64-encoded binary data object

Required: No

DataType

Amazon SNS supports the following logical data types: String, String.Array, Number, and Binary. For more information, see Message Attribute Data Types.

Type: String

Required: Yes

StringValue

Strings are Unicode with UTF8 binary encoding. For a list of code values, see ASCII Printable Characters.

Type: String

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2

PlatformApplication

Platform application object.

Contents

Attributes , Attributes.entry.N.key (key), Attributes.entry.N.value (value)

Attributes for platform application object.

Type: String to string map

Required: No
PlatformApplicationArn

PlatformApplicationArn for platform application object.

Type: String

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2

Subscription

A wrapper type for the attributes of an Amazon SNS subscription.

Contents

Endpoint

The subscription's endpoint (format depends on the protocol).

Type: String

Required: No

Owner

The subscription's owner.

Type: String

Required: No

Protocol

The subscription's protocol.

Type: String

Required: No

SubscriptionArn

The subscription's ARN.

Type: String

Required: No

TopicArn

The ARN of the subscription's topic.

Type: String

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2

Tag

The list of tags to be added to the specified topic.

Contents

Key

The required key portion of the tag.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Required: Yes

Value

The optional value portion of the tag.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 256.

Required: Yes

See Also

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

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2

Topic

A wrapper type for the topic's Amazon Resource Name (ARN). To retrieve a topic's attributes, use `GetTopicAttributes`.

Contents

TopicArn

The topic's ARN.

Type: String

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2

Common Parameters

The following list contains the parameters that all actions use for signing Signature Version 4 requests with a query string. Any action-specific parameters are listed in the topic for that action. For more information about Signature Version 4, see Signature Version 4 Signing Process in the *Amazon Web Services General Reference*.

Action

> The action to be performed.
>
> Type: string
>
> Required: Yes

Version

> The API version that the request is written for, expressed in the format YYYY-MM-DD.
>
> Type: string
>
> Required: Yes

X-Amz-Algorithm

> The hash algorithm that you used to create the request signature.
>
> Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.
>
> Type: string
>
> Valid Values: `AWS4-HMAC-SHA256`
>
> Required: Conditional

X-Amz-Credential

> The credential scope value, which is a string that includes your access key, the date, the region you are targeting, the service you are requesting, and a termination string ("aws4_request"). The value is expressed in the following format: *access_key*/*YYYYMMDD*/*region*/*service*/aws4_request.
>
> For more information, see Task 2: Create a String to Sign for Signature Version 4 in the *Amazon Web Services General Reference*.
>
> Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.
>
> Type: string
>
> Required: Conditional

X-Amz-Date

> The date that is used to create the signature. The format must be ISO 8601 basic format (YYYYMMDD'T'HHMMSS'Z'). For example, the following date time is a valid X-Amz-Date value: `20120325T120000Z`.
>
> Condition: X-Amz-Date is optional for all requests; it can be used to override the date used for signing requests. If the Date header is specified in the ISO 8601 basic format, X-Amz-Date is

not required. When X-Amz-Date is used, it always overrides the value of the Date header. For more information, see Handling Dates in Signature Version 4 in the *Amazon Web Services General Reference*.

Type: string

Required: Conditional

X-Amz-Security-Token

The temporary security token that was obtained through a call to AWS Security Token Service (AWS STS). For a list of services that support temporary security credentials from AWS Security Token Service, go to AWS Services That Work with IAM in the *IAM User Guide*.

Condition: If you're using temporary security credentials from the AWS Security Token Service, you must include the security token.

Type: string

Required: Conditional

X-Amz-Signature

Specifies the hex-encoded signature that was calculated from the string to sign and the derived signing key.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

X-Amz-SignedHeaders

Specifies all the HTTP headers that were included as part of the canonical request. For more information about specifying signed headers, see Task 1: Create a Canonical Request For Signature Version 4 in the *Amazon Web Services General Reference*.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

Common Errors

This section lists the errors common to the API actions of all AWS services. For errors specific to an API action for this service, see the topic for that API action.

AccessDeniedException

You do not have sufficient access to perform this action.

HTTP Status Code: 400

IncompleteSignature

The request signature does not conform to AWS standards.

HTTP Status Code: 400

InternalFailure

The request processing has failed because of an unknown error, exception or failure.

HTTP Status Code: 500

InvalidAction

The action or operation requested is invalid. Verify that the action is typed correctly.

HTTP Status Code: 400

InvalidClientTokenId

The X.509 certificate or AWS access key ID provided does not exist in our records.

HTTP Status Code: 403

InvalidParameterCombination

Parameters that must not be used together were used together.

HTTP Status Code: 400

InvalidParameterValue

An invalid or out-of-range value was supplied for the input parameter.

HTTP Status Code: 400

InvalidQueryParameter

The AWS query string is malformed or does not adhere to AWS standards.

HTTP Status Code: 400

MalformedQueryString

The query string contains a syntax error.

HTTP Status Code: 404

MissingAction

The request is missing an action or a required parameter.

HTTP Status Code: 400

MissingAuthenticationToken

The request must contain either a valid (registered) AWS access key ID or X.509 certificate.

HTTP Status Code: 403

MissingParameter

A required parameter for the specified action is not supplied.

HTTP Status Code: 400

OptInRequired

The AWS access key ID needs a subscription for the service.

HTTP Status Code: 403

RequestExpired

The request reached the service more than 15 minutes after the date stamp on the request or more than 15 minutes after the request expiration date (such as for pre-signed URLs), or the date stamp on the request is more than 15 minutes in the future.

HTTP Status Code: 400

ServiceUnavailable

The request has failed due to a temporary failure of the server.

HTTP Status Code: 503

ThrottlingException

The request was denied due to request throttling.

HTTP Status Code: 400

ValidationError

The input fails to satisfy the constraints specified by an AWS service.

HTTP Status Code: 400