AWS Security Token Service

API Reference

API Version 2011-06-15

AWS Security Token Service: API Reference

Table of Contents

Welcome

The AWS Security Token Service is a web service that enables you to request temporary, limited-privilege credentials for AWS Identity and Access Management (IAM) users or for users that you authenticate (federated users). This guide provides descriptions of the AWS Security Token Service API.

For more detailed information about using this service, go to Using Temporary Security Credentials.

For information about setting up signatures and authorization through the API, go to Signing AWS API Requests in the *AWS General Reference*. For general information about the Query API, go to Making Query Requests in *Using IAM*. For information about using security tokens with other AWS products, go to Using Temporary Security Credentials to Access AWS in *Using Temporary Security Credentials*.

If you're new to AWS and need additional technical information about a specific AWS product, you can find the product's technical documentation at http://aws.amazon.com/documentation/.

We will refer to Amazon Identity and Access Management using the abbreviated form IAM. All copyrights and legal protections still apply.

This document was last updated on July 2, 2012.

Actions

The actions described in this guide are called using the AWS Query protocol.

The following actions are supported:

GetFederationToken

Description

The GetFederationToken action returns a set of temporary credentials for a federated user with the user name and policy specified in the request. The credentials consist of an Access Key ID, a Secret Access Key, and a security token. Credentials created by IAM users are valid for the specified duration, between one and 36 hours; credentials created using account credentials last one hour.

The federated user who holds these credentials has any permissions allowed by the intersection of the specified policy and any resource or user policies that apply to the caller of the GetFederationToken API, and any resource policies that apply to the federated user's Amazon Resource Name (ARN). For more information about how token permissions work, see Controlling Permissions in Temporary Credentials in *Using AWS Identity and Access Management*. For information about using GetFederationToken to create temporary credentials, see Creating Temporary Credentials to Enable Access for Federated Users in *Using AWS Identity and Access Management*.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 11).

Name	Description	Required
`DurationSeconds`	The duration, in seconds, that the session should last. Acceptable durations for federation sessions range from 3600s (one hour) to 129600s (36 hours), with 43200s (12 hours) as the default. Type: Integer	No
`Name`	The name of the federated user associated with the credentials. For information about limitations on user names, go to Limitations on IAM Entities in *Using AWS Identity and Access Management*. Type: String Length constraints: Minimum length of 2. Maximum length of 32.	Yes
`Policy`	A policy specifying the permissions to associate with the credentials. The caller can delegate their own permissions by specifying a policy, and both policies will be checked when a service call is made. For more information about how permissions work in the context of temporary credentials, see Controlling Permissions in Temporary Credentials in *Using AWS Identity and Access Management*. Type: String Length constraints: Minimum length of 1. Maximum length of 2048.	No

Response Elements

The following elements come wrapped in a `GetFederationTokenResult` structure.

Name	Description
`Credentials`	Credentials for the service API authentication. Type: Credentials (p. 8)

Name	Description
FederatedUser	Identifiers for the federated user associated with the credentials. You can use the federated user's ARN in your resource policies. Type: FederatedUser (p. 9)
PackedPolicySize	A percentage value indicating the size of the policy in packed form. Policies for which the packed size is greater than 100% of the allowed value are rejected by the service. Type: Integer

Errors

For information about the common errors that all actions use, see Common Errors (p. 13).

Error	Description	HTTP Status Code
MalformedPolicyDocument	The request was rejected because the policy document was malformed. The error message describes the specific error.	400
PackedPolicyTooLarge	The request was rejected because the policy document was too large. The error message describes how big the policy document is, in packed form, as a percentage of what the API allows.	400

Examples

Sample Request

```
https://sts.amazonaws.com/
?Version=2011-06-15
&Action=GetFederationToken
&Name=Bob
&Policy=%7B%22Statement%22%3A%5B%7B%22Sid%22%3A%22Stmt1%22%2C%22Effect%22%
 3A%22Allow%22%2C%22Action%22%3A%22s3%3A*%22%2C%22Resource%22%3A%22*%22%7D
 %5D%7D
&DurationSeconds=3600
&AUTHPARAMS
```

Sample Response

```
<GetFederationTokenResponse xmlns="https://sts.amazonaws.com/doc/
2011-06-15/">
  <GetFederationTokenResult>
    <Credentials>
      <SessionToken>
       AQoDYXdzEPT//////////wEXAMPLEtc764bNrC9SAPBSM22wDOk4x4HIZ8j4FZTwdQW
       LWsKWHGBuFqwAeMicRXmxfpSPfIeoIYRqTflfKD8YUuwthAx7mSEI/qkPpKPi/kMcGd
       QrmGdeehM4IC1NtBmUpp2wUE8phUZampKsburEDy0KPkyQDYwT7WZ0wq5VSXDvp75YU
```

```
       9HFvlRd8Tx6q6fE8YQcHNVXAkiY9q6d+xo0rKwT38xVqr7ZD0u0iPPkUL64lIZbqBAz
       +scqKmlzm8FDrypNC9Yjc8fPOLn9FX9KSYvKTr4rvx3iSIlTJabIQwj2ICCR/oLxBA==
      </SessionToken>
      <SecretAccessKey>
       wJalrXUtnFEMI/K7MDENG/bPxRfiCYzEXAMPLEKEY
      </SecretAccessKey>
      <Expiration>2011-07-15T23:28:33.359Z</Expiration>
      <AccessKeyId>AKIAIOSFODNN7EXAMPLE</AccessKeyId>
    </Credentials>
    <FederatedUser>
      <Arn>arn:aws:sts::123456789012:federated-user/Bob</Arn>
      <FederatedUserId>123456789012:Bob</FederatedUserId>
    </FederatedUser>
    <PackedPolicySize>6</PackedPolicySize>
  </GetFederationTokenResult>
  <ResponseMetadata>
    <RequestId>c6104cbe-af31-11e0-8154-cbc7ccf896c7</RequestId>
  </ResponseMetadata>
</GetFederationTokenResponse>
```

GetSessionToken

Description

The GetSessionToken action returns a set of temporary credentials for an AWS account or IAM user. The credentials consist of an Access Key ID, a Secret Access Key, and a security token. These credentials are valid for the specified duration only. The session duration for IAM users can be between one and 36 hours, with a default of 12 hours. The session duration for AWS account owners is restricted to one hour. Providing the AWS Multi-Factor Authentication (MFA) device serial number and the token code is optional.

For more information about using GetSessionToken to create temporary credentials, go to Creating Temporary Credentials to Enable Access for IAM Users in *Using IAM*.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 11).

Name	Description	Required
`DurationSeconds`	The duration, in seconds, that the credentials should remain valid. Acceptable durations for IAM user sessions range from 3600s (one hour) to 129600s (36 hours), with 43200s (12 hours) as the default. Sessions for AWS account owners are restricted to a maximum of 3600s (one hour). Type: Integer	No
`SerialNumber`	The identification number of the MFA device for the user. If the IAM user has a policy requiring MFA authentication (or is in a group requiring MFA authentication) to access resources, provide the device value here. The value is in the **Security Credentials** tab of the user's details pane in the IAM console. If the IAM user has an active MFA device, the details pane displays a **Multi-Factor Authentication Device** value. The value is either for a virtual device, such as `arn:aws:iam::123456789012:mfa/user`, or it is the device serial number for a hardware device (usually the number from the back of the device), such as `GAHT12345678`. For more information, see Using Multi-Factor Authentication (MFA) Devices with AWS in *Using IAM*. Type: String Length constraints: Minimum length of 9. Maximum length of 256.	No
`TokenCode`	The value provided by the MFA device. If the user has an access policy requiring an MFA code (or is in a group requiring an MFA code), provide the value here to get permission to resources as specified in the access policy. If MFA authentication is required, and the user does not provide a code when requesting a set of temporary security credentials, the user will receive an "access denied" response when requesting resources that require MFA authentication. For more information, see Using Multi-Factor Authentication (MFA) Devices with AWS in *Using IAM*. Type: String Length constraints: Minimum length of 6. Maximum length of 6.	No

Response Elements

The following elements come wrapped in a `GetSessionTokenResult` structure.

Name	Description
`Credentials`	The session credentials for API authentication. Type: Credentials (p. 8)

Examples

Sample Request

```
https://sts.amazonaws.com/
?Version=2011-06-15
&Action=GetSessionToken
&DurationSeconds=3600
&SerialNumber=YourMFADeviceSerialNumber
&TokenCode=123456
&AUTHPARAMS
```

Sample Response

```
<GetSessionTokenResponse xmlns="https://sts.amazonaws.com/doc/2011-06-15/">
  <GetSessionTokenResult>
    <Credentials>
      <SessionToken>
       AQoEXAMPLEH4aoAH0gNCAPyJxz4BlCFFxWNE1OPTgk5TthT+FvwqnKwRcOIfrRh3c/L
       To6UDdyJwOOvEVPvLXCrrrUtdnniCEXAMPLE/IvU1dYUg2RVAJBanLiHb4IgRmpRV3z
       rkuWJOgQs8IZZaIv2BXIa2R4OlgkBN9bkUDNCJiBeb/AXlzBBko7b15fjrBs2+cTQtp
       Z3CYWFXG8C5zqx37wnOE49mRl/+OtkIKGO7fAE
      </SessionToken>
      <SecretAccessKey>
      wJalrXUtnFEMI/K7MDENG/bPxRfiCYzEXAMPLEKEY
      </SecretAccessKey>
      <Expiration>2011-07-11T19:55:29.611Z</Expiration>
      <AccessKeyId>AKIAIOSFODNN7EXAMPLE</AccessKeyId>
    </Credentials>
  </GetSessionTokenResult>
  <ResponseMetadata>
    <RequestId>58c5dbae-abef-11e0-8cfe-09039844ac7d</RequestId>
  </ResponseMetadata>
</GetSessionTokenResponse>
```

Data Types

The AWS Security Token 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 the response is not guaranteed. Applications should not assume a particular order.

The following data types are supported:

Credentials

Description

AWS credentials for API authentication.

Contents

Name	Description
`AccessKeyId`	AccessKeyId ID that identifies the temporary credentials. Type: String Length constraints: Minimum length of 16. Maximum length of 32.
`Expiration`	The date on which these credentials expire. Type: DateTime

Name	Description
SecretAccessKey	The Secret Access Key to sign requests. Type: String
SessionToken	The security token that users must pass to the service API to use the temporary credentials. Type: String

FederatedUser

Description

Identifiers for the federated user associated with the credentials.

Contents

Name	Description
Arn	The ARN specifying the federated user associated with the credentials. For more information about ARNs and how to use them in policies, see Identifiers for IAM Entities in *Using AWS Identity and Access Management*. Type: String Length constraints: Minimum length of 20. Maximum length of 2048.
FederatedUserId	The string identifying the federated user associated with the credentials, similar to the UserId of an IAM user. Type: String Length constraints: Minimum length of 2. Maximum length of 96.

GetFederationTokenResult

Description

Contains the result of a successful invocation of the GetFederationToken (p. 3) action.

Contents

Name	Description
Credentials	Credentials for the service API authentication. Type: Credentials (p. 8)
FederatedUser	Identifiers for the federated user associated with the credentials. You can use the federated user's ARN in your resource policies. Type: FederatedUser (p. 9)

Name	Description
`PackedPolicySize`	A percentage value indicating the size of the policy in packed form. Policies for which the packed size is greater than 100% of the allowed value are rejected by the service. Type: Integer

GetSessionTokenResult

Description

Contains the result of a successful invocation of the GetSessionToken (p. 6) action.

Contents

Name	Description
`Credentials`	The session credentials for API authentication. Type: Credentials (p. 8)

Common Query Parameters

This section lists the request parameters that all actions use. Any action-specific parameters are listed in the topic for the action.

Parameter Name	Description	Required
Action	The action to perform. Default: None Type: String	Yes
AuthParams	The parameters required to authenticate a query request. Contains: AWSAccessKeyID SignatureVersion Timestamp Signature Default: None	Conditional
AWSAccessKeyId	The Access Key ID corresponding to the AWS Secret Access Key you used to sign the request. Default: None Type: String	Yes
Expires	The date and time at which the request signature expires, in the format YYYY-MM-DDThh:mm:ssZ, as specified in the ISO 8601 standard. Condition: Requests must include either *Timestamp* or *Expires*, but not both. Default: None Type: String	Conditional

Parameter Name	Description	Required
SecurityToken	The temporary security token obtained through a call to AWS Security Token Service. For a list of services that support AWS Security Token Service, go to Using Temporary Security Credentials to Access AWS in *Using Temporary Security Credentials*. Default: None Type: String	
Signature	The digital signature you created for the request. Refer to the service's developer documentation for information about how to generate the signature. Default: None Type: String	Yes
SignatureMethod	The hash algorithm you used to create the request signature. Default: None Valid Values: `HmacSHA256` | `HmacSHA1`. Type: String	Yes
SignatureVersion	The signature version you use to sign the request. Set this to the value recommended in your product-specific documentation on security. Default: None Type: String	Yes
Timestamp	The date and time the request was signed, in the format YYYY-MM-DDThh:mm:ssZ, as specified in the ISO 8601 standard. Condition: Requests must include either *Timestamp* or *Expires*, but not both. Default: None Type: String	Conditional
Version	The API version to use, in the format YYYY-MM-DD. Default: None Type: String	Yes

Common Errors

This section lists the common errors that all actions return. Any action-specific errors are listed in the topic for the action.

Error	Description	HTTP Status Code
`IncompleteSignature`	The request signature does not conform to AWS standards.	400
`InternalFailure`	The request processing has failed due to some unknown error, exception or failure.	500
`InvalidAction`	The action or operation requested is invalid.	400
`InvalidClientTokenId`	The X.509 certificate or AWS Access Key ID provided does not exist in our records.	403
`InvalidParameterCombination`	Parameters that must not be used together were used together.	400
`InvalidParameterValue`	A bad or out-of-range value was supplied for the input parameter.	400
`InvalidQueryParameter`	AWS query string is malformed, does not adhere to AWS standards.	400
`MalformedQueryString`	The query string is malformed.	404
`MissingAction`	The request is missing an action or operation parameter.	400
`MissingAuthenticationToken`	Request must contain either a valid (registered) AWS Access Key ID or X.509 certificate.	403
`MissingParameter`	An input parameter that is mandatory for processing the request is not supplied.	400

Error	Description	HTTP Status Code
`OptInRequired`	The AWS Access Key ID needs a subscription for the service.	403
`RequestExpired`	Request is past expires date or the request date (either with 15 minute padding), or the request date occurs more than 15 minutes in the future.	400
`ServiceUnavailable`	The request has failed due to a temporary failure of the server.	503
`Throttling`	Request was denied due to request throttling.	400