AWS SSO Identity Store

API Reference

AWS SSO Identity Store: API Reference

Table of Contents

Welcome 1
Actions 2
DescribeGroup 3
Request Syntax 3
Request Parameters 3
Response Syntax 3
Response Elements 3
Errors 4
See Also 5
DescribeUser 6
Request Syntax 6
Request Parameters 6
Response Syntax 6
Response Elements 6
Errors 7
See Also 7
ListGroups 9
Request Syntax 9
Request Parameters 9
Response Syntax 10
Response Elements 10
Errors 10
See Also 11
ListUsers 12
Request Syntax 12
Request Parameters 12
Response Syntax 13
Response Elements 13
Errors 13
See Also 14
Data Types 15
Filter 16
Contents 16
See Also 16
Group 17
Contents 17
See Also 17
User 18
Contents 18
See Also 18
Common Parameters 19
Common Errors 21
Document History 23

Welcome to the AWS SSO Identity Store API Reference Guide

The AWS Single Sign-On (SSO) Identity Store service provides a single place to retrieve all of your identities (users and groups). For more information about AWS SSO, see the AWS Single Sign-On User Guide.

This guide describes the AWS SSO Identity Store operations that you can call programatically and includes detailed information on data types and errors. Future updates to AWS SSO Identity Store APIs, including additions for creation and modification of users and groups, will be documented in this reference as they are released.

Notes

- The Identity Store API operations were built to support AWS SSO assignment API operations by providing the required identifiers for users and groups. The scope of these API operations is currently limited to only this functionality and does not include generic operations, such as listing all users or groups in the AWS SSO Identity Store.
- AWS provides SDKs that consist of libraries and sample code for various programming languages and platforms (Java, Ruby, .Net, iOS, Android, and more). The SDKs provide a convenient way to create programmatic access to AWS Directory Service and other AWS services. For more information about the AWS SDKs, including how to download and install them, see Tools for Amazon Web Services.

Actions

The following actions are supported:

- DescribeGroup (p. 3)
- DescribeUser (p. 6)
- ListGroups (p. 9)
- ListUsers (p. 12)

DescribeGroup

Retrieves the group metadata and attributes from `GroupId` in an identity store.

Request Syntax

```
{
   "GroupId": "string",
   "IdentityStoreId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 19).

The request accepts the following data in JSON format.

GroupId (p. 3)

The identifier for a group in the identity store.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 47.

Pattern: `^([0-9a-f]{10}-|)[A-Fa-f0-9]{8}-[A-Fa-f0-9]{4}-[A-Fa-f0-9]{4}-[A-Fa-f0-9]{4}-[A-Fa-f0-9]{12}$`

Required: Yes

IdentityStoreId (p. 3)

The globally unique identifier for the identity store, such as `d-1234567890`. In this example, `d-` is a fixed prefix, and `1234567890` is a randomly generated string that contains number and lower case letters. This value is generated at the time that a new identity store is created.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 12.

Pattern: `^d-[0-9a-f]{10}$`

Required: Yes

Response Syntax

```
{
   "DisplayName": "string",
   "GroupId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

DisplayName (p. 3)

Contains the group's display name value. The length limit is 1,024 characters. This value can consist of letters, accented characters, symbols, numbers, punctuation, tab, new line, carriage return, space, and nonbreaking space in this attribute. The characters `<>;:%` are excluded. This value is specified at the time that the group is created and stored as an attribute of the group object in the identity store.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: `[\p{L}\p{M}\p{S}\p{N}\p{P}\t\n\r  ]+`

GroupId (p. 3)

The identifier for a group in the identity store.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 47.

Pattern: `^([0-9a-f]{10}-|)[A-Fa-f0-9]{8}-[A-Fa-f0-9]{4}-[A-Fa-f0-9]{4}-[A-Fa-f0-9]{4}-[A-Fa-f0-9]{12}$`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 21).

AccessDeniedException

You do not have sufficient access to perform this action.

HTTP Status Code: 400

InternalServerException

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

HTTP Status Code: 500

ResourceNotFoundException

Indicates that a requested resource is not found.

HTTP Status Code: 400

ThrottlingException

Indicates that the principal has crossed the throttling limits of the API operations.

HTTP Status Code: 400

ValidationException

The request failed because it contains a syntax error.

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

DescribeUser

Retrieves the user metadata and attributes from `UserId` in an identity store.

Request Syntax

```
{
   "IdentityStoreId": "string",
   "UserId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 19).

The request accepts the following data in JSON format.

IdentityStoreId (p. 6)

The globally unique identifier for the identity store, such as `d-1234567890`. In this example, `d-` is a fixed prefix, and `1234567890` is a randomly generated string that contains number and lower case letters. This value is generated at the time that a new identity store is created.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 12.

Pattern: `^d-[0-9a-f]{10}$`

Required: Yes

UserId (p. 6)

The identifier for a user in the identity store.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 47.

Pattern: `^([0-9a-f]{10}-|)[A-Fa-f0-9]{8}-[A-Fa-f0-9]{4}-[A-Fa-f0-9]{4}-[A-Fa-f0-9]{4}-[A-Fa-f0-9]{12}$`

Required: Yes

Response Syntax

```
{
   "UserId": "string",
   "UserName": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

UserId (p. 6)

The identifier for a user in the identity store.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 47.

Pattern: `^([0-9a-f]{10}-|)[A-Fa-f0-9]{8}-[A-Fa-f0-9]{4}-[A-Fa-f0-9]{4}-[A-Fa-f0-9]{4}-[A-Fa-f0-9]{12}$`

UserName (p. 6)

Contains the user's user name value. The length limit is 128 characters. This value can consist of letters, accented characters, symbols, numbers, and punctuation. The characters `<>;:%` are excluded. This value is specified at the time the user is created and stored as an attribute of the user object in the identity store.

Type: String

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

Pattern: `[\p{L}\p{M}\p{S}\p{N}\p{P}]+`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 21).

AccessDeniedException

You do not have sufficient access to perform this action.

HTTP Status Code: 400

InternalServerException

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

HTTP Status Code: 500

ResourceNotFoundException

Indicates that a requested resource is not found.

HTTP Status Code: 400

ThrottlingException

Indicates that the principal has crossed the throttling limits of the API operations.

HTTP Status Code: 400

ValidationException

The request failed because it contains a syntax error.

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

ListGroups

Lists the attribute name and value of the group that you specified in the search. We only support `DisplayName` as a valid filter attribute path currently, and filter is required. This API returns minimum attributes, including `GroupId` and group `DisplayName` in the response.

Request Syntax

```
{
   "Filters": [
      {
         "AttributePath": "string",
         "AttributeValue": "string"
      }
   ],
   "IdentityStoreId": "string",
   "MaxResults": number,
   "NextToken": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 19).

The request accepts the following data in JSON format.

Filters (p. 9)

A list of `Filter` objects, which is used in the `ListUsers` and `ListGroups` request.

Type: Array of Filter (p. 16) objects

Required: Yes

IdentityStoreId (p. 9)

The globally unique identifier for the identity store, such as `d-1234567890`. In this example, `d-` is a fixed prefix, and `1234567890` is a randomly generated string that contains number and lower case letters. This value is generated at the time that a new identity store is created.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 12.

Pattern: `^d-[0-9a-f]{10}$`

Required: Yes

MaxResults (p. 9)

The maximum number of results to be returned per request. This parameter is used in the `ListUsers` and `ListGroups` request to specify how many results to return in one page. The length limit is 50 characters.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 50.

Required: No

NextToken (p. 9)

The pagination token used for the `ListUsers` and `ListGroups` API operations. This value is generated by the identity store service. It is returned in the API response if the total results are more than the size of one page. This token is also returned when it is used in the API request to search for the next page.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 65535.

Pattern: `^[-a-zA-Z0-9+=/:]*`

Required: No

Response Syntax

```
{
   "Groups": [
      {
         "DisplayName": "string",
         "GroupId": "string"
      }
   ],
   "NextToken": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

Groups (p. 10)

A list of `Group` objects in the identity store.

Type: Array of Group (p. 17) objects

NextToken (p. 10)

The pagination token used for the `ListUsers` and `ListGroups` API operations. This value is generated by the identity store service. It is returned in the API response if the total results are more than the size of one page. This token is also returned when it1 is used in the API request to search for the next page.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 65535.

Pattern: `^[-a-zA-Z0-9+=/:]*`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 21).

AccessDeniedException

You do not have sufficient access to perform this action.

HTTP Status Code: 400

InternalServerException

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

HTTP Status Code: 500

ResourceNotFoundException

Indicates that a requested resource is not found.

HTTP Status Code: 400

ThrottlingException

Indicates that the principal has crossed the throttling limits of the API operations.

HTTP Status Code: 400

ValidationException

The request failed because it contains a syntax error.

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

ListUsers

Lists the attribute name and value of the user that you specified in the search. We only support `UserName` as a valid filter attribute path currently, and filter is required. This API returns minimum attributes, including `UserId` and `UserName` in the response.

Request Syntax

```
{
   "Filters": [
      {
         "AttributePath": "string",
         "AttributeValue": "string"
      }
   ],
   "IdentityStoreId": "string",
   "MaxResults": number,
   "NextToken": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 19).

The request accepts the following data in JSON format.

Filters (p. 12)

A list of `Filter` objects, which is used in the `ListUsers` and `ListGroups` request.

Type: Array of Filter (p. 16) objects

Required: Yes

IdentityStoreId (p. 12)

The globally unique identifier for the identity store, such as `d-1234567890`. In this example, `d-` is a fixed prefix, and `1234567890` is a randomly generated string that contains number and lower case letters. This value is generated at the time that a new identity store is created.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 12.

Pattern: `^d-[0-9a-f]{10}$`

Required: Yes

MaxResults (p. 12)

The maximum number of results to be returned per request. This parameter is used in the `ListUsers` and `ListGroups` request to specify how many results to return in one page. The length limit is 50 characters.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 50.

Required: No

NextToken (p. 12)

The pagination token used for the `ListUsers` and `ListGroups` API operations. This value is generated by the identity store service. It is returned in the API response if the total results are more than the size of one page. This token is also returned when it is used in the API request to search for the next page.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 65535.

Pattern: `^[-a-zA-Z0-9+=/:]*`

Required: No

Response Syntax

```
{
   "NextToken": "string",
   "Users": [
      {
         "UserId": "string",
         "UserName": "string"
      }
   ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

NextToken (p. 13)

The pagination token used for the `ListUsers` and `ListGroups` API operations. This value is generated by the identity store service. It is returned in the API response if the total results are more than the size of one page. This token is also returned when it is used in the API request to search for the next page.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 65535.

Pattern: `^[-a-zA-Z0-9+=/:]*`

Users (p. 13)

A list of `User` objects in the identity store.

Type: Array of User (p. 18) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 21).

AccessDeniedException

You do not have sufficient access to perform this action.

HTTP Status Code: 400

InternalServerException

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

HTTP Status Code: 500

ResourceNotFoundException

Indicates that a requested resource is not found.

HTTP Status Code: 400

ThrottlingException

Indicates that the principal has crossed the throttling limits of the API operations.

HTTP Status Code: 400

ValidationException

The request failed because it contains a syntax error.

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

Data Types

The AWS SSO Identity Store 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:

- Filter (p. 16)
- Group (p. 17)
- User (p. 18)

Filter

A query filter used by `ListUsers` and `ListGroups`. This filter object provides the attribute name and attribute value to search users or groups.

Contents

AttributePath

The attribute path that is used to specify which attribute name to search. Length limit is 255 characters. For example, `UserName` is a valid attribute path for the `ListUsers` API, and `DisplayName` is a valid attribute path for the `ListGroups` API.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `[\p{L}\p{M}\p{S}\p{N}\p{P}  ]+`

Required: Yes

AttributeValue

Represents the data for an attribute. Each attribute value is described as a name-value pair.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: `[\p{L}\p{M}\p{S}\p{N}\p{P}\t\n\r  ]+`

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 V2
- AWS SDK for Ruby V3

Group

A group object, which contains a specified group's metadata and attributes.

Contents

DisplayName

Contains the group's display name value. The length limit is 1,024 characters. This value can consist of letters, accented characters, symbols, numbers, punctuation, tab, new line, carriage return, space, and nonbreaking space in this attribute. The characters `<>;:%` are excluded. This value is specified at the time the group is created and stored as an attribute of the group object in the identity store.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: `[\p{L}\p{M}\p{S}\p{N}\p{P}\t\n\r  ]+`

Required: Yes

GroupId

The identifier for a group in the identity store.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 47.

Pattern: `^([0-9a-f]{10}-|)[A-Fa-f0-9]{8}-[A-Fa-f0-9]{4}-[A-Fa-f0-9]{4}-[A-Fa-f0-9]{4}-[A-Fa-f0-9]{12}$`

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 V2
- AWS SDK for Ruby V3

User

A user object, which contains a specified user's metadata and attributes.

Contents

UserId

The identifier for a user in the identity store.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 47.

Pattern: `^([0-9a-f]{10}-|)[A-Fa-f0-9]{8}-[A-Fa-f0-9]{4}-[A-Fa-f0-9]{4}-[A-Fa-f0-9]{4}-[A-Fa-f0-9]{12}$`

Required: Yes

UserName

Contains the user's user name value. The length limit is 128 characters. This value can consist of letters, accented characters, symbols, numbers, and punctuation. The characters `<>;:%` are excluded. This value is specified at the time the user is created and stored as an attribute of the user object in the identity store.

Type: String

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

Pattern: `[\p{L}\p{M}\p{S}\p{N}\p{P}]+`

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 V2
- AWS SDK for Ruby V3

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

NotAuthorized

You do not have permission to perform this action.

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

Document History

The following table describes the important changes to the documentation in this release of the *AWS SSO Identity Store* API Reference Guide.

- **Latest documentation update:** August 18, 2020

Change	Description	Date Changed
New guide	This is the first release of the AWS SSO Identity Store API Reference Guide.	August 18, 2020