Amazon Cognito Identity Provider

API Reference

API Version 2016-04-18

Amazon Cognito Identity Provider: API Reference

Table of Contents

Welcome

Using the Amazon Cognito User Pools API, you can create a user pool to manage directories and users. You can authenticate a user to obtain tokens related to user identity and access policies.

This API reference provides information about user pools in Amazon Cognito User Pools.

For more information, see the Amazon Cognito Documentation.

This document was last published on November 19, 2018.

Actions

The following actions are supported:

AddCustomAttributes

Adds additional user attributes to the user pool schema.

Request Syntax

```
{
   "CustomAttributes": [
      {
         "AttributeDataType": "string",
         "DeveloperOnlyAttribute": boolean,
         "Mutable": boolean,
         "Name": "string",
         "NumberAttributeConstraints": {
            "MaxValue": "string",
            "MinValue": "string"
         },
         "Required": boolean,
         "StringAttributeConstraints": {
            "MaxLength": "string",
            "MinLength": "string"
         }
      }
   ],
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

CustomAttributes (p. 5)

An array of custom attributes, such as Mutable and Name.

Type: Array of SchemaAttributeType (p. 360) objects

Array Members: Minimum number of 1 item. Maximum number of 25 items.

Required: Yes

UserPoolId (p. 5)

The user pool ID for the user pool where you want to add custom attributes.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserImportInProgressException

This exception is thrown when you are trying to modify a user pool while a user import job is in progress for that pool.

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

AdminAddUserToGroup

Adds the specified user to the specified group.

Requires developer credentials.

Request Syntax

```
{
   "GroupName": "string",
   "Username": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

GroupName (p. 7)

The group name.

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

Username (p. 7)

The username for the user.

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

UserPoolId (p. 7)

The user pool ID for the user pool.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

AdminConfirmSignUp

Confirms user registration as an admin without using a confirmation code. Works on any user.

Requires developer credentials.

Request Syntax

```
{
   "Username": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

Username (p. 9)

The user name for which you want to confirm user registration.

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

UserPoolId (p. 9)

The user pool ID for which you want to confirm user registration.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidLambdaResponseException

This exception is thrown when the Amazon Cognito service encounters an invalid AWS Lambda response.

HTTP Status Code: 400

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

LimitExceededException

This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyFailedAttemptsException

This exception is thrown when the user has made too many failed attempts for a given action (e.g., sign in).

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UnexpectedLambdaException

This exception is thrown when the Amazon Cognito service encounters an unexpected exception with the AWS Lambda service.

HTTP Status Code: 400

UserLambdaValidationException

This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

AdminCreateUser

Creates a new user in the specified user pool.

If `MessageAction` is not set, the default is to send a welcome message via email or phone (SMS).

> **Note**
> This message is based on a template that you configured in your call to CreateUserPool (p. 113) or UpdateUserPool (p. 288). This template includes your custom sign-up instructions and placeholders for user name and temporary password.

Alternatively, you can call AdminCreateUser with "SUPPRESS" for the `MessageAction` parameter, and Amazon Cognito will not send any email.

In either case, the user will be in the `FORCE_CHANGE_PASSWORD` state until they sign in and change their password.

AdminCreateUser requires developer credentials.

Request Syntax

```
{
   "DesiredDeliveryMediums": [ "string" ],
   "ForceAliasCreation": boolean,
   "MessageAction": "string",
   "TemporaryPassword": "string",
   "UserAttributes": [
      {
         "Name": "string",
         "Value": "string"
      }
   ],
   "Username": "string",
   "UserPoolId": "string",
   "ValidationData": [
      {
         "Name": "string",
         "Value": "string"
      }
   ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

DesiredDeliveryMediums (p. 12)

Specify `"EMAIL"` if email will be used to send the welcome message. Specify `"SMS"` if the phone number will be used. The default value is `"SMS"`. More than one value can be specified.

Type: Array of strings

Valid Values: `SMS | EMAIL`

Required: No

ForceAliasCreation (p. 12)

This parameter is only used if the `phone_number_verified` or `email_verified` attribute is set to `True`. Otherwise, it is ignored.

If this parameter is set to `True` and the phone number or email address specified in the UserAttributes parameter already exists as an alias with a different user, the API call will migrate the alias from the previous user to the newly created user. The previous user will no longer be able to log in using that alias.

If this parameter is set to `False`, the API throws an `AliasExistsException` error if the alias already exists. The default value is `False`.

Type: Boolean

Required: No

MessageAction (p. 12)

Set to `"RESEND"` to resend the invitation message to a user that already exists and reset the expiration limit on the user's account. Set to `"SUPPRESS"` to suppress sending the message. Only one value can be specified.

Type: String

Valid Values: `RESEND | SUPPRESS`

Required: No

TemporaryPassword (p. 12)

The user's temporary password. This password must conform to the password policy that you specified when you created the user pool.

The temporary password is valid only once. To complete the Admin Create User flow, the user must enter the temporary password in the sign-in page along with a new password to be used in all future sign-ins.

This parameter is not required. If you do not specify a value, Amazon Cognito generates one for you.

The temporary password can only be used until the user account expiration limit that you specified when you created the user pool. To reset the account after that time limit, you must call `AdminCreateUser` again, specifying `"RESEND"` for the `MessageAction` parameter.

Type: String

Length Constraints: Minimum length of 6. Maximum length of 256.

Pattern: `[\S]+`

Required: No

UserAttributes (p. 12)

An array of name-value pairs that contain user attributes and attribute values to be set for the user to be created. You can create a user without specifying any attributes other than `Username`. However, any attributes that you specify as required (in CreateUserPool (p. 113) or in the **Attributes** tab of the console) must be supplied either by you (in your call to `AdminCreateUser`) or by the user (when he or she signs up in response to your welcome message).

For custom attributes, you must prepend the `custom:` prefix to the attribute name.

To send a message inviting the user to sign up, you must specify the user's email address or phone number. This can be done in your call to AdminCreateUser or in the **Users** tab of the Amazon Cognito console for managing your user pools.

In your call to `AdminCreateUser`, you can set the `email_verified` attribute to `True`, and you can set the `phone_number_verified` attribute to `True`. (You can also do this by calling AdminUpdateUserAttributes (p. 79).)

- **email**: The email address of the user to whom the message that contains the code and username will be sent. Required if the `email_verified` attribute is set to `True`, or if `"EMAIL"` is specified in the `DesiredDeliveryMediums` parameter.
- **phone_number**: The phone number of the user to whom the message that contains the code and username will be sent. Required if the `phone_number_verified` attribute is set to `True`, or if `"SMS"` is specified in the `DesiredDeliveryMediums` parameter.

Type: Array of AttributeType (p. 315) objects

Required: No

Username (p. 12)

The username for the user. Must be unique within the user pool. Must be a UTF-8 string between 1 and 128 characters. After the user is created, the username cannot be changed.

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

UserPoolId (p. 12)

The user pool ID for the user pool where the user will be created.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

ValidationData (p. 12)

The user's validation data. This is an array of name-value pairs that contain user attributes and attribute values that you can use for custom validation, such as restricting the types of user accounts that can be registered. For example, you might choose to allow or disallow user sign-up based on the user's domain.

To configure custom validation, you must create a Pre Sign-up Lambda trigger for the user pool as described in the Amazon Cognito Developer Guide. The Lambda trigger receives the validation data and uses it in the validation process.

The user's validation data is not persisted.

Type: Array of AttributeType (p. 315) objects

Required: No

Response Syntax

```
{
   "User": {
      "Attributes": [
```

```
            {
               "Name": "string",
               "Value": "string"
            }
         ],
         "Enabled": boolean,
         "MFAOptions": [
            {
               "AttributeName": "string",
               "DeliveryMedium": "string"
            }
         ],
         "UserCreateDate": number,
         "UserLastModifiedDate": number,
         "Username": "string",
         "UserStatus": "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.

User (p. 14)

The newly created user.

Type: UserType (p. 388) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

CodeDeliveryFailureException

This exception is thrown when a verification code fails to deliver successfully.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidLambdaResponseException

This exception is thrown when the Amazon Cognito service encounters an invalid AWS Lambda response.

HTTP Status Code: 400

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidPasswordException

This exception is thrown when the Amazon Cognito service encounters an invalid password.

HTTP Status Code: 400

InvalidSmsRoleAccessPolicyException

This exception is returned when the role provided for SMS configuration does not have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is invalid for the role provided for SMS configuration. This can happen if you do not trust **cognito-idp.amazonaws.com** or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

PreconditionNotMetException

This exception is thrown when a precondition is not met.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UnexpectedLambdaException

This exception is thrown when the Amazon Cognito service encounters an unexpected exception with the AWS Lambda service.

HTTP Status Code: 400

UnsupportedUserStateException

The request failed because the user is in an unsupported state.

HTTP Status Code: 400

UserLambdaValidationException

This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.

HTTP Status Code: 400

UsernameExistsException

This exception is thrown when Amazon Cognito encounters a user name that already exists in the user pool.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

AdminDeleteUser

Deletes a user as an administrator. Works on any user.

Requires developer credentials.

Request Syntax

```
{
   "Username": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

Username (p. 18)

The user name of the user you wish to delete.

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

UserPoolId (p. 18)

The user pool ID for the user pool where you want to delete the user.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

AdminDeleteUserAttributes

Deletes the user attributes in a user pool as an administrator. Works on any user.

Requires developer credentials.

Request Syntax

```
{
   "UserAttributeNames": [ "string" ],
   "Username": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

UserAttributeNames (p. 20)

An array of strings representing the user attribute names you wish to delete.

For custom attributes, you must prepend the `custom:` prefix to the attribute name.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 32.

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

Required: Yes

Username (p. 20)

The user name of the user from which you would like to delete attributes.

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

UserPoolId (p. 20)

The user pool ID for the user pool where you want to delete user attributes.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

AdminDisableProviderForUser

Disables the user from signing in with the specified external (SAML or social) identity provider. If the user to disable is a Cognito User Pools native username + password user, they are not permitted to use their password to sign-in. If the user to disable is a linked external IdP user, any link between that user and an existing user is removed. The next time the external user (no longer attached to the previously linked `DestinationUser`) signs in, they must create a new user account. See AdminLinkProviderForUser (p. 44).

This action is enabled only for admin access and requires developer credentials.

The `ProviderName` must match the value specified when creating an IdP for the pool.

To disable a native username + password user, the `ProviderName` value must be `Cognito` and the `ProviderAttributeName` must be `Cognito_Subject`, with the `ProviderAttributeValue` being the name that is used in the user pool for the user.

The `ProviderAttributeName` must always be `Cognito_Subject` for social identity providers. The `ProviderAttributeValue` must always be the exact subject that was used when the user was originally linked as a source user.

For de-linking a SAML identity, there are two scenarios. If the linked identity has not yet been used to sign-in, the `ProviderAttributeName` and `ProviderAttributeValue` must be the same values that were used for the `SourceUser` when the identities were originally linked in the AdminLinkProviderForUser (p. 44) call. (If the linking was done with `ProviderAttributeName` set to `Cognito_Subject`, the same applies here). However, if the user has already signed in, the `ProviderAttributeName` must be `Cognito_Subject` and `ProviderAttributeValue` must be the subject of the SAML assertion.

Request Syntax

```
{
   "User": {
      "ProviderAttributeName": "string",
      "ProviderAttributeValue": "string",
      "ProviderName": "string"
   },
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

User (p. 22)

The user to be disabled.

Type: ProviderUserIdentifierType (p. 353) object

Required: Yes

UserPoolId (p. 22)

The user pool ID for the user pool.

Type: String

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

AliasExistsException

This exception is thrown when a user tries to confirm the account with an email or phone number that has already been supplied as an alias from a different account. This exception tells user that an account with this email or phone already exists.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

AdminDisableUser

Disables the specified user as an administrator. Works on any user.

Requires developer credentials.

Request Syntax

```
{
   "Username": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

Username (p. 25)

The user name of the user you wish to disable.

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

UserPoolId (p. 25)

The user pool ID for the user pool where you want to disable the user.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

AdminEnableUser

Enables the specified user as an administrator. Works on any user.

Requires developer credentials.

Request Syntax

```
{
   "Username": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

Username (p. 27)

The user name of the user you wish to enable.

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

UserPoolId (p. 27)

The user pool ID for the user pool where you want to enable the user.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

AdminForgetDevice

Forgets the device, as an administrator.

Requires developer credentials.

Request Syntax

```
{
   "DeviceKey": "string",
   "Username": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

DeviceKey (p. 29)

The device key.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-f-]+`

Required: Yes

Username (p. 29)

The user name.

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

UserPoolId (p. 29)

The user pool ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidUserPoolConfigurationException

This exception is thrown when the user pool configuration is invalid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

AdminGetDevice

Gets the device, as an administrator.

Requires developer credentials.

Request Syntax

```
{
   "DeviceKey": "string",
   "Username": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

DeviceKey (p. 31)

The device key.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-f-]+`

Required: Yes

Username (p. 31)

The user name.

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

UserPoolId (p. 31)

The user pool ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
```

```
   "Device": {
      "DeviceAttributes": [
         {
            "Name": "string",
            "Value": "string"
         }
      ],
      "DeviceCreateDate": number,
      "DeviceKey": "string",
      "DeviceLastAuthenticatedDate": number,
      "DeviceLastModifiedDate": number
   }
}
```

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.

Device (p. 31)

The device.

Type: DeviceType (p. 328) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidUserPoolConfigurationException

This exception is thrown when the user pool configuration is invalid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

AdminGetUser

Gets the specified user by user name in a user pool as an administrator. Works on any user.

Requires developer credentials.

Request Syntax

```
{
   "Username": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

Username (p. 34)

The user name of the user you wish to retrieve.

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

UserPoolId (p. 34)

The user pool ID for the user pool where you want to get information about the user.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "Enabled": boolean,
   "MFAOptions": [
      {
         "AttributeName": "string",
         "DeliveryMedium": "string"
      }
   ],
   "PreferredMfaSetting": "string",
   "UserAttributes": [
      {
```

```
            "Name": "string",
            "Value": "string"
        }
    ],
    "UserCreateDate": number,
    "UserLastModifiedDate": number,
    "UserMFASettingList": [ "string" ],
    "Username": "string",
    "UserStatus": "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.

Enabled (p. 34)

Indicates that the status is enabled.

Type: Boolean

MFAOptions (p. 34)

Specifies the options for MFA (e.g., email or phone number).

Type: Array of MFAOptionType (p. 344) objects

PreferredMfaSetting (p. 34)

The user's preferred MFA setting.

Type: String

UserAttributes (p. 34)

An array of name-value pairs representing user attributes.

Type: Array of AttributeType (p. 315) objects

UserCreateDate (p. 34)

The date the user was created.

Type: Timestamp

UserLastModifiedDate (p. 34)

The date the user was last modified.

Type: Timestamp

UserMFASettingList (p. 34)

The list of the user's MFA settings.

Type: Array of strings

Username (p. 34)

The user name of the user about whom you are receiving information.

Type: String

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

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

UserStatus (p. 34)

The user status. Can be one of the following:

- UNCONFIRMED - User has been created but not confirmed.
- CONFIRMED - User has been confirmed.
- ARCHIVED - User is no longer active.
- COMPROMISED - User is disabled due to a potential security threat.
- UNKNOWN - User status is not known.

Type: String

Valid Values: `UNCONFIRMED | CONFIRMED | ARCHIVED | COMPROMISED | UNKNOWN | RESET_REQUIRED | FORCE_CHANGE_PASSWORD`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

AdminInitiateAuth

Initiates the authentication flow, as an administrator.

Requires developer credentials.

Request Syntax

```
{
   "AnalyticsMetadata": {
      "AnalyticsEndpointId": "string"
   },
   "AuthFlow": "string",
   "AuthParameters": {
      "string" : "string"
   },
   "ClientId": "string",
   "ClientMetadata": {
      "string" : "string"
   },
   "ContextData": {
      "EncodedData": "string",
      "HttpHeaders": [
         {
            "headerName": "string",
            "headerValue": "string"
         }
      ],
      "IpAddress": "string",
      "ServerName": "string",
      "ServerPath": "string"
   },
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AnalyticsMetadata (p. 38)

The analytics metadata for collecting Amazon Pinpoint metrics for `AdminInitiateAuth` calls.

Type: AnalyticsMetadataType (p. 314) object

Required: No

AuthFlow (p. 38)

The authentication flow for this call to execute. The API action will depend on this value. For example:

- `REFRESH_TOKEN_AUTH` will take in a valid refresh token and return new tokens.
- `USER_SRP_AUTH` will take in `USERNAME` and `SRP_A` and return the SRP variables to be used for next challenge execution.
- `USER_PASSWORD_AUTH` will take in `USERNAME` and `PASSWORD` and return the next challenge or tokens.

Valid values include:

- `USER_SRP_AUTH`: Authentication flow for the Secure Remote Password (SRP) protocol.
- `REFRESH_TOKEN_AUTH/REFRESH_TOKEN`: Authentication flow for refreshing the access token and ID token by supplying a valid refresh token.
- `CUSTOM_AUTH`: Custom authentication flow.
- `ADMIN_NO_SRP_AUTH`: Non-SRP authentication flow; you can pass in the USERNAME and PASSWORD directly if the flow is enabled for calling the app client.
- `USER_PASSWORD_AUTH`: Non-SRP authentication flow; USERNAME and PASSWORD are passed directly. If a user migration Lambda trigger is set, this flow will invoke the user migration Lambda if the USERNAME is not found in the user pool.

Type: String

Valid Values: `USER_SRP_AUTH | REFRESH_TOKEN_AUTH | REFRESH_TOKEN | CUSTOM_AUTH | ADMIN_NO_SRP_AUTH | USER_PASSWORD_AUTH`

Required: Yes

AuthParameters (p. 38)

The authentication parameters. These are inputs corresponding to the `AuthFlow` that you are invoking. The required values depend on the value of `AuthFlow`:

- For `USER_SRP_AUTH`: `USERNAME` (required), `SRP_A` (required), `SECRET_HASH` (required if the app client is configured with a client secret), `DEVICE_KEY`
- For `REFRESH_TOKEN_AUTH/REFRESH_TOKEN`: `REFRESH_TOKEN` (required), `SECRET_HASH` (required if the app client is configured with a client secret), `DEVICE_KEY`
- For `ADMIN_NO_SRP_AUTH`: `USERNAME` (required), `SECRET_HASH` (if app client is configured with client secret), `PASSWORD` (required), `DEVICE_KEY`
- For `CUSTOM_AUTH`: `USERNAME` (required), `SECRET_HASH` (if app client is configured with client secret), `DEVICE_KEY`

Type: String to string map

Required: No

ClientId (p. 38)

The app client ID.

Type: String

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

Pattern: `[\w+]+`

Required: Yes

ClientMetadata (p. 38)

This is a random key-value pair map which can contain any key and will be passed to your PreAuthentication Lambda trigger as-is. It can be used to implement additional validations around authentication.

Type: String to string map

Required: No

ContextData (p. 38)

Contextual data such as the user's device fingerprint, IP address, or location used for evaluating the risk of an unexpected event by Amazon Cognito advanced security.

Type: ContextDataType (p. 324) object

Required: No

UserPoolId (p. 38)

The ID of the Amazon Cognito user pool.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "AuthenticationResult": {
      "AccessToken": "string",
      "ExpiresIn": number,
      "IdToken": "string",
      "NewDeviceMetadata": {
         "DeviceGroupKey": "string",
         "DeviceKey": "string"
      },
      "RefreshToken": "string",
      "TokenType": "string"
   },
   "ChallengeName": "string",
   "ChallengeParameters": {
      "string" : "string"
   },
   "Session": "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.

AuthenticationResult (p. 40)

The result of the authentication response. This is only returned if the caller does not need to pass another challenge. If the caller does need to pass another challenge before it gets tokens, `ChallengeName`, `ChallengeParameters`, and `Session` are returned.

Type: AuthenticationResultType (p. 316) object

ChallengeName (p. 40)

The name of the challenge which you are responding to with this call. This is returned to you in the `AdminInitiateAuth` response if you need to pass another challenge.

- `MFA_SETUP`: If MFA is required, users who do not have at least one of the MFA methods set up are presented with an `MFA_SETUP` challenge. The user must set up at least one MFA type to continue to authenticate.
- `SELECT_MFA_TYPE`: Selects the MFA type. Valid MFA options are `SMS_MFA` for text SMS MFA, and `SOFTWARE_TOKEN_MFA` for TOTP software token MFA.

- `SMS_MFA`: Next challenge is to supply an `SMS_MFA_CODE`, delivered via SMS.
- `PASSWORD_VERIFIER`: Next challenge is to supply `PASSWORD_CLAIM_SIGNATURE`, `PASSWORD_CLAIM_SECRET_BLOCK`, and `TIMESTAMP` after the client-side SRP calculations.
- `CUSTOM_CHALLENGE`: This is returned if your custom authentication flow determines that the user should pass another challenge before tokens are issued.
- `DEVICE_SRP_AUTH`: If device tracking was enabled on your user pool and the previous challenges were passed, this challenge is returned so that Amazon Cognito can start tracking this device.
- `DEVICE_PASSWORD_VERIFIER`: Similar to `PASSWORD_VERIFIER`, but for devices only.
- `ADMIN_NO_SRP_AUTH`: This is returned if you need to authenticate with `USERNAME` and `PASSWORD` directly. An app client must be enabled to use this flow.
- `NEW_PASSWORD_REQUIRED`: For users which are required to change their passwords after successful first login. This challenge should be passed with `NEW_PASSWORD` and any other required attributes.

Type: String

Valid Values: `SMS_MFA | SOFTWARE_TOKEN_MFA | SELECT_MFA_TYPE | MFA_SETUP | PASSWORD_VERIFIER | CUSTOM_CHALLENGE | DEVICE_SRP_AUTH | DEVICE_PASSWORD_VERIFIER | ADMIN_NO_SRP_AUTH | NEW_PASSWORD_REQUIRED`

ChallengeParameters (p. 40)

The challenge parameters. These are returned to you in the `AdminInitiateAuth` response if you need to pass another challenge. The responses in this parameter should be used to compute inputs to the next call (`AdminRespondToAuthChallenge`).

All challenges require `USERNAME` and `SECRET_HASH` (if applicable).

The value of the `USER_ID_FOR_SRP` attribute will be the user's actual username, not an alias (such as email address or phone number), even if you specified an alias in your call to `AdminInitiateAuth`. This is because, in the `AdminRespondToAuthChallenge` API `ChallengeResponses`, the `USERNAME` attribute cannot be an alias.

Type: String to string map

Session (p. 40)

The session which should be passed both ways in challenge-response calls to the service. If `AdminInitiateAuth` or `AdminRespondToAuthChallenge` API call determines that the caller needs to go through another challenge, they return a session with other challenge parameters. This session should be passed as it is to the next `AdminRespondToAuthChallenge` API call.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidLambdaResponseException

This exception is thrown when the Amazon Cognito service encounters an invalid AWS Lambda response.

HTTP Status Code: 400

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidSmsRoleAccessPolicyException

This exception is returned when the role provided for SMS configuration does not have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is invalid for the role provided for SMS configuration. This can happen if you do not trust **cognito-idp.amazonaws.com** or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.

HTTP Status Code: 400

InvalidUserPoolConfigurationException

This exception is thrown when the user pool configuration is invalid.

HTTP Status Code: 400

MFAMethodNotFoundException

This exception is thrown when Amazon Cognito cannot find a multi-factor authentication (MFA) method.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UnexpectedLambdaException

This exception is thrown when the Amazon Cognito service encounters an unexpected exception with the AWS Lambda service.

HTTP Status Code: 400

UserLambdaValidationException

This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user is not confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

AdminLinkProviderForUser

Links an existing user account in a user pool (`DestinationUser`) to an identity from an external identity provider (`SourceUser`) based on a specified attribute name and value from the external identity provider. This allows you to create a link from the existing user account to an external federated user identity that has not yet been used to sign in, so that the federated user identity can be used to sign in as the existing user account.

For example, if there is an existing user with a username and password, this API links that user to a federated user identity, so that when the federated user identity is used, the user signs in as the existing user account.

> **Important**
> Because this API allows a user with an external federated identity to sign in as an existing user in the user pool, it is critical that it only be used with external identity providers and provider attributes that have been trusted by the application owner.

See also AdminDisableProviderForUser (p. 22).

This action is enabled only for admin access and requires developer credentials.

Request Syntax

```
{
   "DestinationUser": {
      "ProviderAttributeName": "string",
      "ProviderAttributeValue": "string",
      "ProviderName": "string"
   },
   "SourceUser": {
      "ProviderAttributeName": "string",
      "ProviderAttributeValue": "string",
      "ProviderName": "string"
   },
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

DestinationUser (p. 44)

The existing user in the user pool to be linked to the external identity provider user account. Can be a native (Username + Password) Cognito User Pools user or a federated user (for example, a SAML or Facebook user). If the user doesn't exist, an exception is thrown. This is the user that is returned when the new user (with the linked identity provider attribute) signs in.

For a native username + password user, the `ProviderAttributeValue` for the `DestinationUser` should be the username in the user pool. For a federated user, it should be the provider-specific `user_id`.

The `ProviderAttributeName` of the `DestinationUser` is ignored.

The `ProviderName` should be set to `Cognito` for users in Cognito user pools.

Type: ProviderUserIdentifierType (p. 353) object

Required: Yes

SourceUser (p. 44)

An external identity provider account for a user who does not currently exist yet in the user pool. This user must be a federated user (for example, a SAML or Facebook user), not another native user.

If the `SourceUser` is a federated social identity provider user (Facebook, Google, or Login with Amazon), you must set the `ProviderAttributeName` to `Cognito_Subject`. For social identity providers, the `ProviderName` will be `Facebook`, `Google`, or `LoginWithAmazon`, and Cognito will automatically parse the Facebook, Google, and Login with Amazon tokens for `id`, `sub`, and `user_id`, respectively. The `ProviderAttributeValue` for the user must be the same value as the `id`, `sub`, or `user_id` value found in the social identity provider token.

For SAML, the `ProviderAttributeName` can be any value that matches a claim in the SAML assertion. If you wish to link SAML users based on the subject of the SAML assertion, you should map the subject to a claim through the SAML identity provider and submit that claim name as the `ProviderAttributeName`. If you set `ProviderAttributeName` to `Cognito_Subject`, Cognito will automatically parse the default unique identifier found in the subject from the SAML token.

Type: ProviderUserIdentifierType (p. 353) object

Required: Yes

UserPoolId (p. 44)

The user pool ID for the user pool.

Type: String

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

AliasExistsException

This exception is thrown when a user tries to confirm the account with an email or phone number that has already been supplied as an alias from a different account. This exception tells user that an account with this email or phone already exists.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

AdminListDevices

Lists devices, as an administrator.

Requires developer credentials.

Request Syntax

```
{
   "Limit": number,
   "PaginationToken": "string",
   "Username": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

Limit (p. 47)

The limit of the devices request.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 60.

Required: No

PaginationToken (p. 47)

The pagination token.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

Required: No

Username (p. 47)

The user name.

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

UserPoolId (p. 47)

The user pool ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "Devices": [
      {
         "DeviceAttributes": [
            {
               "Name": "string",
               "Value": "string"
            }
         ],
         "DeviceCreateDate": number,
         "DeviceKey": "string",
         "DeviceLastAuthenticatedDate": number,
         "DeviceLastModifiedDate": number
      }
   ],
   "PaginationToken": "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.

Devices (p. 48)

The devices in the list of devices response.

Type: Array of DeviceType (p. 328) objects

PaginationToken (p. 48)

The pagination token.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidUserPoolConfigurationException

This exception is thrown when the user pool configuration is invalid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

AdminListGroupsForUser

Lists the groups that the user belongs to.

Requires developer credentials.

Request Syntax

```
{
   "Limit": number,
   "NextToken": "string",
   "Username": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

Limit (p. 50)

The limit of the request to list groups.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 60.

Required: No

NextToken (p. 50)

An identifier that was returned from the previous call to this operation, which can be used to return the next set of items in the list.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

Required: No

Username (p. 50)

The username for the user.

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

UserPoolId (p. 50)

The user pool ID for the user pool.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "Groups": [
      {
         "CreationDate": number,
         "Description": "string",
         "GroupName": "string",
         "LastModifiedDate": number,
         "Precedence": number,
         "RoleArn": "string",
         "UserPoolId": "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. 51)

The groups that the user belongs to.

Type: Array of GroupType (p. 335) objects

NextToken (p. 51)

An identifier that was returned from the previous call to this operation, which can be used to return the next set of items in the list.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

AdminListUserAuthEvents

Lists a history of user activity and any risks detected as part of Amazon Cognito advanced security.

Request Syntax

```
{
   "MaxResults": number,
   "NextToken": "string",
   "Username": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

MaxResults (p. 53)

The maximum number of authentication events to return.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 60.

Required: No

NextToken (p. 53)

A pagination token.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

Required: No

Username (p. 53)

The user pool username or an alias.

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

UserPoolId (p. 53)

The user pool ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "AuthEvents": [
      {
         "ChallengeResponses": [
            {
               "ChallengeName": "string",
               "ChallengeResponse": "string"
            }
         ],
         "CreationDate": number,
         "EventContextData": {
            "City": "string",
            "Country": "string",
            "DeviceName": "string",
            "IpAddress": "string",
            "Timezone": "string"
         },
         "EventFeedback": {
            "FeedbackDate": number,
            "FeedbackValue": "string",
            "Provider": "string"
         },
         "EventId": "string",
         "EventResponse": "string",
         "EventRisk": {
            "RiskDecision": "string",
            "RiskLevel": "string"
         },
         "EventType": "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.

AuthEvents (p. 54)

The response object. It includes the `EventID`, `EventType`, `CreationDate`, `EventRisk`, and `EventResponse`.

Type: Array of AuthEventType (p. 318) objects

NextToken (p. 54)

A pagination token.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

HTTP Status Code: 400

UserPoolAddOnNotEnabledException

This exception is thrown when user pool add-ons are not enabled.

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

AdminRemoveUserFromGroup

Removes the specified user from the specified group.

Requires developer credentials.

Request Syntax

```
{
   "GroupName": "string",
   "Username": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

GroupName (p. 57)

The group name.

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

Username (p. 57)

The username for the user.

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

UserPoolId (p. 57)

The user pool ID for the user pool.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

AdminResetUserPassword

Resets the specified user's password in a user pool as an administrator. Works on any user.

When a developer calls this API, the current password is invalidated, so it must be changed. If a user tries to sign in after the API is called, the app will get a PasswordResetRequiredException exception back and should direct the user down the flow to reset the password, which is the same as the forgot password flow. In addition, if the user pool has phone verification selected and a verified phone number exists for the user, or if email verification is selected and a verified email exists for the user, calling this API will also result in sending a message to the end user with the code to change their password.

Requires developer credentials.

Request Syntax

```
{
   "Username": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

Username (p. 59)

The user name of the user whose password you wish to reset.

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

UserPoolId (p. 59)

The user pool ID for the user pool where you want to reset the user's password.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidEmailRoleAccessPolicyException

This exception is thrown when Amazon Cognito is not allowed to use your email identity. HTTP status code: 400.

HTTP Status Code: 400

InvalidLambdaResponseException

This exception is thrown when the Amazon Cognito service encounters an invalid AWS Lambda response.

HTTP Status Code: 400

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidSmsRoleAccessPolicyException

This exception is returned when the role provided for SMS configuration does not have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is invalid for the role provided for SMS configuration. This can happen if you do not trust **cognito-idp.amazonaws.com** or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.

HTTP Status Code: 400

LimitExceededException

This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UnexpectedLambdaException

This exception is thrown when the Amazon Cognito service encounters an unexpected exception with the AWS Lambda service.

HTTP Status Code: 400

UserLambdaValidationException

This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

AdminRespondToAuthChallenge

Responds to an authentication challenge, as an administrator.

Requires developer credentials.

Request Syntax

```
{
   "AnalyticsMetadata": {
      "AnalyticsEndpointId": "string"
   },
   "ChallengeName": "string",
   "ChallengeResponses": {
      "string" : "string"
   },
   "ClientId": "string",
   "ContextData": {
      "EncodedData": "string",
      "HttpHeaders": [
         {
            "headerName": "string",
            "headerValue": "string"
         }
      ],
      "IpAddress": "string",
      "ServerName": "string",
      "ServerPath": "string"
   },
   "Session": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AnalyticsMetadata (p. 62)

The analytics metadata for collecting Amazon Pinpoint metrics for `AdminRespondToAuthChallenge` calls.

Type: AnalyticsMetadataType (p. 314) object

Required: No

ChallengeName (p. 62)

The challenge name. For more information, see AdminInitiateAuth (p. 38).

Type: String

Valid Values: `SMS_MFA | SOFTWARE_TOKEN_MFA | SELECT_MFA_TYPE | MFA_SETUP | PASSWORD_VERIFIER | CUSTOM_CHALLENGE | DEVICE_SRP_AUTH | DEVICE_PASSWORD_VERIFIER | ADMIN_NO_SRP_AUTH | NEW_PASSWORD_REQUIRED`

Required: Yes

ChallengeResponses (p. 62)

The challenge responses. These are inputs corresponding to the value of `ChallengeName`, for example:

- `SMS_MFA`: `SMS_MFA_CODE`, `USERNAME`, `SECRET_HASH` (if app client is configured with client secret).
- `PASSWORD_VERIFIER`: `PASSWORD_CLAIM_SIGNATURE`, `PASSWORD_CLAIM_SECRET_BLOCK`, `TIMESTAMP`, `USERNAME`, `SECRET_HASH` (if app client is configured with client secret).
- `ADMIN_NO_SRP_AUTH`: `PASSWORD`, `USERNAME`, `SECRET_HASH` (if app client is configured with client secret).
- `NEW_PASSWORD_REQUIRED`: `NEW_PASSWORD`, any other required attributes, `USERNAME`, `SECRET_HASH` (if app client is configured with client secret).

The value of the `USERNAME` attribute must be the user's actual username, not an alias (such as email address or phone number). To make this easier, the `AdminInitiateAuth` response includes the actual username value in the `USERNAMEUSER_ID_FOR_SRP` attribute, even if you specified an alias in your call to `AdminInitiateAuth`.

Type: String to string map

Required: No

ClientId (p. 62)

The app client ID.

Type: String

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

Pattern: `[\w+]+`

Required: Yes

ContextData (p. 62)

Contextual data such as the user's device fingerprint, IP address, or location used for evaluating the risk of an unexpected event by Amazon Cognito advanced security.

Type: ContextDataType (p. 324) object

Required: No

Session (p. 62)

The session which should be passed both ways in challenge-response calls to the service. If `InitiateAuth` or `RespondToAuthChallenge` API call determines that the caller needs to go through another challenge, they return a session with other challenge parameters. This session should be passed as it is to the next `RespondToAuthChallenge` API call.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Required: No

UserPoolId (p. 62)

The ID of the Amazon Cognito user pool.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "AuthenticationResult": {
      "AccessToken": "string",
      "ExpiresIn": number,
      "IdToken": "string",
      "NewDeviceMetadata": {
         "DeviceGroupKey": "string",
         "DeviceKey": "string"
      },
      "RefreshToken": "string",
      "TokenType": "string"
   },
   "ChallengeName": "string",
   "ChallengeParameters": {
      "string" : "string"
   },
   "Session": "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.

AuthenticationResult (p. 64)

The result returned by the server in response to the authentication request.

Type: AuthenticationResultType (p. 316) object

ChallengeName (p. 64)

The name of the challenge. For more information, see AdminInitiateAuth (p. 38).

Type: String

Valid Values: `SMS_MFA | SOFTWARE_TOKEN_MFA | SELECT_MFA_TYPE | MFA_SETUP | PASSWORD_VERIFIER | CUSTOM_CHALLENGE | DEVICE_SRP_AUTH | DEVICE_PASSWORD_VERIFIER | ADMIN_NO_SRP_AUTH | NEW_PASSWORD_REQUIRED`

ChallengeParameters (p. 64)

The challenge parameters. For more information, see AdminInitiateAuth (p. 38).

Type: String to string map

Session (p. 64)

The session which should be passed both ways in challenge-response calls to the service. If the InitiateAuth (p. 201) or RespondToAuthChallenge (p. 238) API call determines that the caller needs to go through another challenge, they return a session with other challenge parameters. This session should be passed as it is to the next `RespondToAuthChallenge` API call.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

AliasExistsException

This exception is thrown when a user tries to confirm the account with an email or phone number that has already been supplied as an alias from a different account. This exception tells user that an account with this email or phone already exists.

HTTP Status Code: 400

CodeMismatchException

This exception is thrown if the provided code does not match what the server was expecting.

HTTP Status Code: 400

ExpiredCodeException

This exception is thrown if a code has expired.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidLambdaResponseException

This exception is thrown when the Amazon Cognito service encounters an invalid AWS Lambda response.

HTTP Status Code: 400

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidPasswordException

This exception is thrown when the Amazon Cognito service encounters an invalid password.

HTTP Status Code: 400

InvalidSmsRoleAccessPolicyException

This exception is returned when the role provided for SMS configuration does not have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is invalid for the role provided for SMS configuration. This can happen if you do not trust **cognito-idp.amazonaws.com** or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.

HTTP Status Code: 400

InvalidUserPoolConfigurationException

This exception is thrown when the user pool configuration is invalid.

HTTP Status Code: 400

MFAMethodNotFoundException

This exception is thrown when Amazon Cognito cannot find a multi-factor authentication (MFA) method.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

SoftwareTokenMFANotFoundException

This exception is thrown when the software token TOTP multi-factor authentication (MFA) is not enabled for the user pool.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UnexpectedLambdaException

This exception is thrown when the Amazon Cognito service encounters an unexpected exception with the AWS Lambda service.

HTTP Status Code: 400

UserLambdaValidationException

This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user is not confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

AdminSetUserMFAPreference

Sets the user's multi-factor authentication (MFA) preference.

Request Syntax

```
{
   "SMSMfaSettings": {
      "Enabled": boolean,
      "PreferredMfa": boolean
   },
   "SoftwareTokenMfaSettings": {
      "Enabled": boolean,
      "PreferredMfa": boolean
   },
   "Username": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

SMSMfaSettings (p. 68)

The SMS text message MFA settings.

Type: SMSMfaSettingsType (p. 364) object

Required: No

SoftwareTokenMfaSettings (p. 68)

The time-based one-time password software token MFA settings.

Type: SoftwareTokenMfaSettingsType (p. 366) object

Required: No

Username (p. 68)

The user pool username or alias.

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

UserPoolId (p. 68)

The user pool ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user is not confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

AdminSetUserSettings

Sets all the user settings for a specified user name. Works on any user.

Requires developer credentials.

Request Syntax

```
{
   "MFAOptions": [
      {
         "AttributeName": "string",
         "DeliveryMedium": "string"
      }
   ],
   "Username": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

MFAOptions (p. 71)

Specifies the options for MFA (e.g., email or phone number).

Type: Array of MFAOptionType (p. 344) objects

Required: Yes

Username (p. 71)

The user name of the user for whom you wish to set user settings.

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

UserPoolId (p. 71)

The user pool ID for the user pool where you want to set the user's settings, such as MFA options.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

AdminUpdateAuthEventFeedback

Provides feedback for an authentication event as to whether it was from a valid user. This feedback is used for improving the risk evaluation decision for the user pool as part of Amazon Cognito advanced security.

Request Syntax

```
{
   "EventId": "string",
   "FeedbackValue": "string",
   "Username": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

EventId (p. 73)

The authentication event ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 50.

Pattern: `[\w+-]+`

Required: Yes

FeedbackValue (p. 73)

The authentication event feedback value.

Type: String

Valid Values: `Valid | Invalid`

Required: Yes

Username (p. 73)

The user pool username.

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

UserPoolId (p. 73)

The user pool ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

HTTP Status Code: 400

UserPoolAddOnNotEnabledException

This exception is thrown when user pool add-ons are not enabled.

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

AdminUpdateDeviceStatus

Updates the device status as an administrator.

Requires developer credentials.

Request Syntax

```
{
   "DeviceKey": "string",
   "DeviceRememberedStatus": "string",
   "Username": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

DeviceKey (p. 76)

The device key.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-f-]+`

Required: Yes

DeviceRememberedStatus (p. 76)

The status indicating whether a device has been remembered or not.

Type: String

Valid Values: `remembered | not_remembered`

Required: No

Username (p. 76)

The user name.

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

UserPoolId (p. 76)

The user pool ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidUserPoolConfigurationException

This exception is thrown when the user pool configuration is invalid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

AdminUpdateUserAttributes

Updates the specified user's attributes, including developer attributes, as an administrator. Works on any user.

For custom attributes, you must prepend the `custom:` prefix to the attribute name.

In addition to updating user attributes, this API can also be used to mark phone and email as verified.

Requires developer credentials.

Request Syntax

```
{
   "UserAttributes": [
      {
         "Name": "string",
         "Value": "string"
      }
   ],
   "Username": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

UserAttributes (p. 79)

An array of name-value pairs representing user attributes.

For custom attributes, you must prepend the `custom:` prefix to the attribute name.

Type: Array of AttributeType (p. 315) objects

Required: Yes

Username (p. 79)

The user name of the user for whom you want to update user attributes.

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

UserPoolId (p. 79)

The user pool ID for the user pool where you want to update user attributes.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

AliasExistsException

This exception is thrown when a user tries to confirm the account with an email or phone number that has already been supplied as an alias from a different account. This exception tells user that an account with this email or phone already exists.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidLambdaResponseException

This exception is thrown when the Amazon Cognito service encounters an invalid AWS Lambda response.

HTTP Status Code: 400

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UnexpectedLambdaException

This exception is thrown when the Amazon Cognito service encounters an unexpected exception with the AWS Lambda service.

HTTP Status Code: 400

UserLambdaValidationException

This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

AdminUserGlobalSignOut

Signs out users from all devices, as an administrator.

Requires developer credentials.

Request Syntax

```
{
   "Username": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

Username (p. 82)

The user name.

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

UserPoolId (p. 82)

The user pool ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

AssociateSoftwareToken

Returns a unique generated shared secret key code for the user account. The request takes an access token or a session string, but not both.

Request Syntax

```
{
   "AccessToken": "string",
   "Session": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AccessToken (p. 84)

The access token.

Type: String

Pattern: `[A-Za-z0-9-_=.]+`

Required: No

Session (p. 84)

The session which should be passed both ways in challenge-response calls to the service. This allows authentication of the user as part of the MFA setup process.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Required: No

Response Syntax

```
{
   "SecretCode": "string",
   "Session": "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.

SecretCode (p. 84)

A unique generated shared secret code that is used in the TOTP algorithm to generate a one time code.

Type: String

Length Constraints: Minimum length of 16.

Pattern: `[A-Za-z0-9]+`

Session (p. 84)

The session which should be passed both ways in challenge-response calls to the service. This allows authentication of the user as part of the MFA setup process.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

SoftwareTokenMFANotFoundException

This exception is thrown when the software token TOTP multi-factor authentication (MFA) is not enabled for the user pool.

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

ChangePassword

Changes the password for a specified user in a user pool.

Request Syntax

```
{
   "AccessToken": "string",
   "PreviousPassword": "string",
   "ProposedPassword": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AccessToken (p. 87)

The access token.

Type: String

Pattern: `[A-Za-z0-9-_=.]+`

Required: Yes

PreviousPassword (p. 87)

The old password.

Type: String

Length Constraints: Minimum length of 6. Maximum length of 256.

Pattern: `[\S]+`

Required: Yes

ProposedPassword (p. 87)

The new password.

Type: String

Length Constraints: Minimum length of 6. Maximum length of 256.

Pattern: `[\S]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidPasswordException

This exception is thrown when the Amazon Cognito service encounters an invalid password.

HTTP Status Code: 400

LimitExceededException

This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user is not confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

ConfirmDevice

Confirms tracking of the device. This API call is the call that begins device tracking.

Request Syntax

```
{
   "AccessToken": "string",
   "DeviceKey": "string",
   "DeviceName": "string",
   "DeviceSecretVerifierConfig": {
      "PasswordVerifier": "string",
      "Salt": "string"
   }
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AccessToken (p. 90)

The access token.

Type: String

Pattern: `[A-Za-z0-9-_=.]+`

Required: Yes

DeviceKey (p. 90)

The device key.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-f-]+`

Required: Yes

DeviceName (p. 90)

The device name.

Type: String

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

Required: No

DeviceSecretVerifierConfig (p. 90)

The configuration of the device secret verifier.

Type: DeviceSecretVerifierConfigType (p. 327) object

Required: No

Response Syntax

```
{
   "UserConfirmationNecessary": boolean
}
```

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.

UserConfirmationNecessary (p. 91)

Indicates whether the user confirmation is necessary to confirm the device response.

Type: Boolean

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidLambdaResponseException

This exception is thrown when the Amazon Cognito service encounters an invalid AWS Lambda response.

HTTP Status Code: 400

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidPasswordException

This exception is thrown when the Amazon Cognito service encounters an invalid password.

HTTP Status Code: 400

InvalidUserPoolConfigurationException

This exception is thrown when the user pool configuration is invalid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UsernameExistsException

This exception is thrown when Amazon Cognito encounters a user name that already exists in the user pool.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user is not confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

ConfirmForgotPassword

Allows a user to enter a confirmation code to reset a forgotten password.

Request Syntax

```
{
   "AnalyticsMetadata": {
      "AnalyticsEndpointId": "string"
   },
   "ClientId": "string",
   "ConfirmationCode": "string",
   "Password": "string",
   "SecretHash": "string",
   "UserContextData": {
      "EncodedData": "string"
   },
   "Username": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AnalyticsMetadata (p. 93)

The Amazon Pinpoint analytics metadata for collecting metrics for `ConfirmForgotPassword` calls.

Type: AnalyticsMetadataType (p. 314) object

Required: No

ClientId (p. 93)

The app client ID of the app associated with the user pool.

Type: String

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

Pattern: `[\w+]+`

Required: Yes

ConfirmationCode (p. 93)

The confirmation code sent by a user's request to retrieve a forgotten password. For more information, see ForgotPassword (p. 169)

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `[\S]+`

Required: Yes

Password (p. 93)

The password sent by a user's request to retrieve a forgotten password.

Type: String

Length Constraints: Minimum length of 6. Maximum length of 256.

Pattern: `[\S]+`

Required: Yes

SecretHash (p. 93)

A keyed-hash message authentication code (HMAC) calculated using the secret key of a user pool client and username plus the client ID in the message.

Type: String

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

Pattern: `[\w+=/]+`

Required: No

UserContextData (p. 93)

Contextual data such as the user's device fingerprint, IP address, or location used for evaluating the risk of an unexpected event by Amazon Cognito advanced security.

Type: UserContextDataType (p. 370) object

Required: No

Username (p. 93)

The user name of the user for whom you want to enter a code to retrieve a forgotten password.

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

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

CodeMismatchException

This exception is thrown if the provided code does not match what the server was expecting.

HTTP Status Code: 400

ExpiredCodeException

This exception is thrown if a code has expired.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidLambdaResponseException

This exception is thrown when the Amazon Cognito service encounters an invalid AWS Lambda response.

HTTP Status Code: 400

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidPasswordException

This exception is thrown when the Amazon Cognito service encounters an invalid password.

HTTP Status Code: 400

LimitExceededException

This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyFailedAttemptsException

This exception is thrown when the user has made too many failed attempts for a given action (e.g., sign in).

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UnexpectedLambdaException

This exception is thrown when the Amazon Cognito service encounters an unexpected exception with the AWS Lambda service.

HTTP Status Code: 400

UserLambdaValidationException

This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user is not confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

ConfirmSignUp

Confirms registration of a user and handles the existing alias from a previous user.

Request Syntax

```
{
   "AnalyticsMetadata": {
      "AnalyticsEndpointId": "string"
   },
   "ClientId": "string",
   "ConfirmationCode": "string",
   "ForceAliasCreation": boolean,
   "SecretHash": "string",
   "UserContextData": {
      "EncodedData": "string"
   },
   "Username": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AnalyticsMetadata (p. 97)

The Amazon Pinpoint analytics metadata for collecting metrics for `ConfirmSignUp` calls.

Type: AnalyticsMetadataType (p. 314) object

Required: No

ClientId (p. 97)

The ID of the app client associated with the user pool.

Type: String

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

Pattern: `[\w+]+`

Required: Yes

ConfirmationCode (p. 97)

The confirmation code sent by a user's request to confirm registration.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `[\S]+`

Required: Yes

ForceAliasCreation (p. 97)

Boolean to be specified to force user confirmation irrespective of existing alias. By default set to `False`. If this parameter is set to `True` and the phone number/email used for sign up confirmation already exists as an alias with a different user, the API call will migrate the alias from the previous user to the newly created user being confirmed. If set to `False`, the API will throw an **AliasExistsException** error.

Type: Boolean

Required: No

SecretHash (p. 97)

A keyed-hash message authentication code (HMAC) calculated using the secret key of a user pool client and username plus the client ID in the message.

Type: String

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

Pattern: `[\w+=/]+`

Required: No

UserContextData (p. 97)

Contextual data such as the user's device fingerprint, IP address, or location used for evaluating the risk of an unexpected event by Amazon Cognito advanced security.

Type: UserContextDataType (p. 370) object

Required: No

Username (p. 97)

The user name of the user whose registration you wish to confirm.

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

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

AliasExistsException

This exception is thrown when a user tries to confirm the account with an email or phone number that has already been supplied as an alias from a different account. This exception tells user that an account with this email or phone already exists.

HTTP Status Code: 400

CodeMismatchException

This exception is thrown if the provided code does not match what the server was expecting.

HTTP Status Code: 400

ExpiredCodeException

This exception is thrown if a code has expired.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidLambdaResponseException

This exception is thrown when the Amazon Cognito service encounters an invalid AWS Lambda response.

HTTP Status Code: 400

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

LimitExceededException

This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyFailedAttemptsException

This exception is thrown when the user has made too many failed attempts for a given action (e.g., sign in).

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UnexpectedLambdaException

This exception is thrown when the Amazon Cognito service encounters an unexpected exception with the AWS Lambda service.

HTTP Status Code: 400

UserLambdaValidationException

This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

CreateGroup

Creates a new group in the specified user pool.

Requires developer credentials.

Request Syntax

```
{
   "Description": "string",
   "GroupName": "string",
   "Precedence": number,
   "RoleArn": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

Description (p. 101)

A string containing the description of the group.

Type: String

Length Constraints: Maximum length of 2048.

Required: No

GroupName (p. 101)

The name of the group. Must be unique.

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

Precedence (p. 101)

A nonnegative integer value that specifies the precedence of this group relative to the other groups that a user can belong to in the user pool. Zero is the highest precedence value. Groups with lower `Precedence` values take precedence over groups with higher or null `Precedence` values. If a user belongs to two or more groups, it is the group with the lowest precedence value whose role ARN will be used in the `cognito:roles` and `cognito:preferred_role` claims in the user's tokens.

Two groups can have the same `Precedence` value. If this happens, neither group takes precedence over the other. If two groups with the same `Precedence` have the same role ARN, that role is used in the `cognito:preferred_role` claim in tokens for users in each group. If the two groups have different role ARNs, the `cognito:preferred_role` claim is not set in users' tokens.

The default `Precedence` value is null.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

RoleArn (p. 101)

The role ARN for the group.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:([\w+=/,.@-]*)?:[0-9]+:[\w+=/,.@-]+(:[\w+=/,.@-]+)?(:[\w+=/,.@-]+)?`

Required: No

UserPoolId (p. 101)

The user pool ID for the user pool.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "Group": {
      "CreationDate": number,
      "Description": "string",
      "GroupName": "string",
      "LastModifiedDate": number,
      "Precedence": number,
      "RoleArn": "string",
      "UserPoolId": "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.

Group (p. 102)

The group object for the group.

Type: GroupType (p. 335) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

GroupExistsException

This exception is thrown when Amazon Cognito encounters a group that already exists in the user pool.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

LimitExceededException

This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

CreateIdentityProvider

Creates an identity provider for a user pool.

Request Syntax

```
{
   "AttributeMapping": {
      "string" : "string"
   },
   "IdpIdentifiers": [ "string" ],
   "ProviderDetails": {
      "string" : "string"
   },
   "ProviderName": "string",
   "ProviderType": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AttributeMapping (p. 104)

A mapping of identity provider attributes to standard and custom user pool attributes.

Type: String to string map

Key Length Constraints: Minimum length of 1. Maximum length of 32.

Required: No

IdpIdentifiers (p. 104)

A list of identity provider identifiers.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 50 items.

Length Constraints: Minimum length of 1. Maximum length of 40.

Pattern: `[\w\s+=.@-]+`

Required: No

ProviderDetails (p. 104)

The identity provider details, such as `MetadataURL` and `MetadataFile`.

Type: String to string map

Required: Yes

ProviderName (p. 104)

The identity provider name.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 32.

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

Required: Yes

ProviderType (p. 104)

The identity provider type.

Type: String

Valid Values: `SAML | Facebook | Google | LoginWithAmazon | OIDC`

Required: Yes

UserPoolId (p. 104)

The user pool ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "IdentityProvider": {
      "AttributeMapping": {
         "string" : "string"
      },
      "CreationDate": number,
      "IdpIdentifiers": [ "string" ],
      "LastModifiedDate": number,
      "ProviderDetails": {
         "string" : "string"
      },
      "ProviderName": "string",
      "ProviderType": "string",
      "UserPoolId": "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.

IdentityProvider (p. 105)

The newly created identity provider object.

Type: IdentityProviderType (p. 338) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

DuplicateProviderException

This exception is thrown when the provider is already supported by the user pool.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

LimitExceededException

This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

CreateResourceServer

Creates a new OAuth2.0 resource server and defines custom scopes in it.

Request Syntax

```
{
   "Identifier": "string",
   "Name": "string",
   "Scopes": [
      {
         "ScopeDescription": "string",
         "ScopeName": "string"
      }
   ],
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

Identifier (p. 107)

A unique resource server identifier for the resource server. This could be an HTTPS endpoint where the resource server is located. For example, `https://my-weather-api.example.com`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `[\x21\x23-\x5B\x5D-\x7E]+`

Required: Yes

Name (p. 107)

A friendly name for the resource server.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `[\w\s+=,.@-]+`

Required: Yes

Scopes (p. 107)

A list of scopes. Each scope is map, where the keys are `name` and `description`.

Type: Array of ResourceServerScopeType (p. 354) objects

Array Members: Maximum number of 25 items.

Required: No

UserPoolId (p. 107)

The user pool ID for the user pool.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "ResourceServer": {
      "Identifier": "string",
      "Name": "string",
      "Scopes": [
         {
            "ScopeDescription": "string",
            "ScopeName": "string"
         }
      ],
      "UserPoolId": "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.

ResourceServer (p. 108)

The newly created resource server.

Type: ResourceServerType (p. 355) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

LimitExceededException

This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

CreateUserImportJob

Creates the user import job.

Request Syntax

```
{
   "CloudWatchLogsRoleArn": "string",
   "JobName": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

CloudWatchLogsRoleArn (p. 110)

The role ARN for the Amazon CloudWatch Logging role for the user import job.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:([\w+=/,.@-]*)?:[0-9]+:[\w+=/,.@-]+(:[\w+=/,.@-]+)?(:[\w+=/,.@-]+)?`

Required: Yes

JobName (p. 110)

The job name for the user import job.

Type: String

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

Pattern: `[\w\s+=,.@-]+`

Required: Yes

UserPoolId (p. 110)

The user pool ID for the user pool that the users are being imported into.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
```

```
   "UserImportJob": {
      "CloudWatchLogsRoleArn": "string",
      "CompletionDate": number,
      "CompletionMessage": "string",
      "CreationDate": number,
      "FailedUsers": number,
      "ImportedUsers": number,
      "JobId": "string",
      "JobName": "string",
      "PreSignedUrl": "string",
      "SkippedUsers": number,
      "StartDate": number,
      "Status": "string",
      "UserPoolId": "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.

UserImportJob (p. 110)

The job object that represents the user import job.

Type: UserImportJobType (p. 371) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

LimitExceededException

This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

PreconditionNotMetException

This exception is thrown when a precondition is not met.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

CreateUserPool

Creates a new Amazon Cognito user pool and sets the password policy for the pool.

Request Syntax

```
{
   "AdminCreateUserConfig": {
      "AllowAdminCreateUserOnly": boolean,
      "InviteMessageTemplate": {
         "EmailMessage": "string",
         "EmailSubject": "string",
         "SMSMessage": "string"
      },
      "UnusedAccountValidityDays": number
   },
   "AliasAttributes": [ "string" ],
   "AutoVerifiedAttributes": [ "string" ],
   "DeviceConfiguration": {
      "ChallengeRequiredOnNewDevice": boolean,
      "DeviceOnlyRememberedOnUserPrompt": boolean
   },
   "EmailConfiguration": {
      "ReplyToEmailAddress": "string",
      "SourceArn": "string"
   },
   "EmailVerificationMessage": "string",
   "EmailVerificationSubject": "string",
   "LambdaConfig": {
      "CreateAuthChallenge": "string",
      "CustomMessage": "string",
      "DefineAuthChallenge": "string",
      "PostAuthentication": "string",
      "PostConfirmation": "string",
      "PreAuthentication": "string",
      "PreSignUp": "string",
      "PreTokenGeneration": "string",
      "UserMigration": "string",
      "VerifyAuthChallengeResponse": "string"
   },
   "MfaConfiguration": "string",
   "Policies": {
      "PasswordPolicy": {
         "MinimumLength": number,
         "RequireLowercase": boolean,
         "RequireNumbers": boolean,
         "RequireSymbols": boolean,
         "RequireUppercase": boolean
      }
   },
   "PoolName": "string",
   "Schema": [
      {
         "AttributeDataType": "string",
         "DeveloperOnlyAttribute": boolean,
         "Mutable": boolean,
         "Name": "string",
         "NumberAttributeConstraints": {
            "MaxValue": "string",
            "MinValue": "string"
         },
         "Required": boolean,
         "StringAttributeConstraints": {
```

```
            "MaxLength": "string",
            "MinLength": "string"
         }
      }
   ],
   "SmsAuthenticationMessage": "string",
   "SmsConfiguration": {
      "ExternalId": "string",
      "SnsCallerArn": "string"
   },
   "SmsVerificationMessage": "string",
   "UsernameAttributes": [ "string" ],
   "UserPoolAddOns": {
      "AdvancedSecurityMode": "string"
   },
   "UserPoolTags": {
      "string" : "string"
   },
   "VerificationMessageTemplate": {
      "DefaultEmailOption": "string",
      "EmailMessage": "string",
      "EmailMessageByLink": "string",
      "EmailSubject": "string",
      "EmailSubjectByLink": "string",
      "SmsMessage": "string"
   }
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AdminCreateUserConfig (p. 113)

The configuration for `AdminCreateUser` requests.

Type: AdminCreateUserConfigType (p. 312) object

Required: No

AliasAttributes (p. 113)

Attributes supported as an alias for this user pool. Possible values: **phone_number**, **email**, or **preferred_username**.

Type: Array of strings

Valid Values: `phone_number | email | preferred_username`

Required: No

AutoVerifiedAttributes (p. 113)

The attributes to be auto-verified. Possible values: **email**, **phone_number**.

Type: Array of strings

Valid Values: `phone_number | email`

Required: No

DeviceConfiguration (p. 113)

The device configuration.

Type: DeviceConfigurationType (p. 326) object

Required: No

EmailConfiguration (p. 113)

The email configuration.

Type: EmailConfigurationType (p. 331) object

Required: No

EmailVerificationMessage (p. 113)

A string representing the email verification message.

Type: String

Length Constraints: Minimum length of 6. Maximum length of 20000.

Pattern: `[\p{L}\p{M}\p{S}\p{N}\p{P}\s*]*\{####\}[\p{L}\p{M}\p{S}\p{N}\p{P}\s*]*`

Required: No

EmailVerificationSubject (p. 113)

A string representing the email verification subject.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 140.

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

Required: No

LambdaConfig (p. 113)

The Lambda trigger configuration information for the new user pool.

> **Note**
> In a push model, event sources (such as Amazon S3 and custom applications) need permission to invoke a function. So you will need to make an extra call to add permission for these event sources to invoke your Lambda function.
> For more information on using the Lambda API to add permission, see AddPermission .
> For adding permission using the AWS CLI, see add-permission .

Type: LambdaConfigType (p. 340) object

Required: No

MfaConfiguration (p. 113)

Specifies MFA configuration details.

Type: String

Valid Values: `OFF | ON | OPTIONAL`

Required: No

Policies (p. 113)

The policies associated with the new user pool.

Type: UserPoolPolicyType (p. 382) object

Required: No

PoolName (p. 113)

A string used to name the user pool.

Type: String

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

Pattern: `[\w\s+=,.@-]+`

Required: Yes

Schema (p. 113)

An array of schema attributes for the new user pool. These attributes can be standard or custom attributes.

Type: Array of SchemaAttributeType (p. 360) objects

Array Members: Minimum number of 1 item. Maximum number of 50 items.

Required: No

SmsAuthenticationMessage (p. 113)

A string representing the SMS authentication message.

Type: String

Length Constraints: Minimum length of 6. Maximum length of 140.

Pattern: `.*\{####\}.*`

Required: No

SmsConfiguration (p. 113)

The SMS configuration.

Type: SmsConfigurationType (p. 362) object

Required: No

SmsVerificationMessage (p. 113)

A string representing the SMS verification message.

Type: String

Length Constraints: Minimum length of 6. Maximum length of 140.

Pattern: `.*\{####\}.*`

Required: No

UsernameAttributes (p. 113)

Specifies whether email addresses or phone numbers can be specified as usernames when a user signs up.

Type: Array of strings

Valid Values: `phone_number | email`

Required: No

UserPoolAddOns (p. 113)

Used to enable advanced security risk detection. Set the key `AdvancedSecurityMode` to the value "AUDIT".

Type: UserPoolAddOnsType (p. 374) object

Required: No

UserPoolTags (p. 113)

The cost allocation tags for the user pool. For more information, see Adding Cost Allocation Tags to Your User Pool

Type: String to string map

Required: No

VerificationMessageTemplate (p. 113)

The template for the verification message that the user sees when the app requests permission to access the user's information.

Type: VerificationMessageTemplateType (p. 390) object

Required: No

Response Syntax

```
{
   "UserPool": {
      "AdminCreateUserConfig": {
         "AllowAdminCreateUserOnly": boolean,
         "InviteMessageTemplate": {
            "EmailMessage": "string",
            "EmailSubject": "string",
            "SMSMessage": "string"
         },
         "UnusedAccountValidityDays": number
      },
      "AliasAttributes": [ "string" ],
      "Arn": "string",
      "AutoVerifiedAttributes": [ "string" ],
      "CreationDate": number,
      "CustomDomain": "string",
      "DeviceConfiguration": {
         "ChallengeRequiredOnNewDevice": boolean,
         "DeviceOnlyRememberedOnUserPrompt": boolean
      },
      "Domain": "string",
      "EmailConfiguration": {
         "ReplyToEmailAddress": "string",
         "SourceArn": "string"
      },
      "EmailConfigurationFailure": "string",
      "EmailVerificationMessage": "string",
      "EmailVerificationSubject": "string",
```

```
"EstimatedNumberOfUsers": number,
"Id": "string",
"LambdaConfig": {
   "CreateAuthChallenge": "string",
   "CustomMessage": "string",
   "DefineAuthChallenge": "string",
   "PostAuthentication": "string",
   "PostConfirmation": "string",
   "PreAuthentication": "string",
   "PreSignUp": "string",
   "PreTokenGeneration": "string",
   "UserMigration": "string",
   "VerifyAuthChallengeResponse": "string"
},
"LastModifiedDate": number,
"MfaConfiguration": "string",
"Name": "string",
"Policies": {
   "PasswordPolicy": {
      "MinimumLength": number,
      "RequireLowercase": boolean,
      "RequireNumbers": boolean,
      "RequireSymbols": boolean,
      "RequireUppercase": boolean
   }
},
"SchemaAttributes": [
   {
      "AttributeDataType": "string",
      "DeveloperOnlyAttribute": boolean,
      "Mutable": boolean,
      "Name": "string",
      "NumberAttributeConstraints": {
         "MaxValue": "string",
         "MinValue": "string"
      },
      "Required": boolean,
      "StringAttributeConstraints": {
         "MaxLength": "string",
         "MinLength": "string"
      }
   }
],
"SmsAuthenticationMessage": "string",
"SmsConfiguration": {
   "ExternalId": "string",
   "SnsCallerArn": "string"
},
"SmsConfigurationFailure": "string",
"SmsVerificationMessage": "string",
"Status": "string",
"UsernameAttributes": [ "string" ],
"UserPoolAddOns": {
   "AdvancedSecurityMode": "string"
},
"UserPoolTags": {
   "string" : "string"
},
"VerificationMessageTemplate": {
   "DefaultEmailOption": "string",
   "EmailMessage": "string",
   "EmailMessageByLink": "string",
   "EmailSubject": "string",
   "EmailSubjectByLink": "string",
   "SmsMessage": "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.

UserPool (p. 117)

A container for the user pool details.

Type: UserPoolType (p. 383) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidEmailRoleAccessPolicyException

This exception is thrown when Amazon Cognito is not allowed to use your email identity. HTTP status code: 400.

HTTP Status Code: 400

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidSmsRoleAccessPolicyException

This exception is returned when the role provided for SMS configuration does not have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is invalid for the role provided for SMS configuration. This can happen if you do not trust **cognito-idp.amazonaws.com** or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.

HTTP Status Code: 400

LimitExceededException

This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserPoolTaggingException

This exception is thrown when a user pool tag cannot be set or updated.

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

CreateUserPoolClient

Creates the user pool client.

Request Syntax

```
{
   "AllowedOAuthFlows": [ "string" ],
   "AllowedOAuthFlowsUserPoolClient": boolean,
   "AllowedOAuthScopes": [ "string" ],
   "AnalyticsConfiguration": {
      "ApplicationId": "string",
      "ExternalId": "string",
      "RoleArn": "string",
      "UserDataShared": boolean
   },
   "CallbackURLs": [ "string" ],
   "ClientName": "string",
   "DefaultRedirectURI": "string",
   "ExplicitAuthFlows": [ "string" ],
   "GenerateSecret": boolean,
   "LogoutURLs": [ "string" ],
   "ReadAttributes": [ "string" ],
   "RefreshTokenValidity": number,
   "SupportedIdentityProviders": [ "string" ],
   "UserPoolId": "string",
   "WriteAttributes": [ "string" ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AllowedOAuthFlows (p. 121)

Set to `code` to initiate a code grant flow, which provides an authorization code as the response. This code can be exchanged for access tokens with the token endpoint.

Set to `token` to specify that the client should get the access token (and, optionally, ID token, based on scopes) directly.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 3 items.

Valid Values: `code | implicit | client_credentials`

Required: No

AllowedOAuthFlowsUserPoolClient (p. 121)

Set to `True` if the client is allowed to follow the OAuth protocol when interacting with Cognito user pools.

Type: Boolean

Required: No

AllowedOAuthScopes (p. 121)

A list of allowed `OAuth` scopes. Currently supported values are `"phone"`, `"email"`, `"openid"`, and `"Cognito"`.

Type: Array of strings

Array Members: Maximum number of 25 items.

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `[\x21\x23-\x5B\x5D-\x7E]+`

Required: No

AnalyticsConfiguration (p. 121)

The Amazon Pinpoint analytics configuration for collecting metrics for this user pool.

Type: AnalyticsConfigurationType (p. 313) object

Required: No

CallbackURLs (p. 121)

A list of allowed redirect (callback) URLs for the identity providers.

A redirect URI must:

- Be an absolute URI.
- Be registered with the authorization server.
- Not include a fragment component.

See OAuth 2.0 - Redirection Endpoint.

Amazon Cognito requires HTTPS over HTTP except for http://localhost for testing purposes only.

App callback URLs such as myapp://example are also supported.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 100 items.

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

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

Required: No

ClientName (p. 121)

The client name for the user pool client you would like to create.

Type: String

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

Pattern: `[\w\s+=,.@-]+`

Required: Yes

DefaultRedirectURI (p. 121)

The default redirect URI. Must be in the `CallbackURLs` list.

A redirect URI must:

- Be an absolute URI.
- Be registered with the authorization server.
- Not include a fragment component.

See OAuth 2.0 - Redirection Endpoint.

Amazon Cognito requires HTTPS over HTTP except for http://localhost for testing purposes only.

App callback URLs such as myapp://example are also supported.

Type: String

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

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

Required: No

ExplicitAuthFlows (p. 121)

The explicit authentication flows.

Type: Array of strings

Valid Values: `ADMIN_NO_SRP_AUTH | CUSTOM_AUTH_FLOW_ONLY | USER_PASSWORD_AUTH`

Required: No

GenerateSecret (p. 121)

Boolean to specify whether you want to generate a secret for the user pool client being created.

Type: Boolean

Required: No

LogoutURLs (p. 121)

A list of allowed logout URLs for the identity providers.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 100 items.

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

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

Required: No

ReadAttributes (p. 121)

The read attributes.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: No

RefreshTokenValidity (p. 121)

The time limit, in days, after which the refresh token is no longer valid and cannot be used.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 3650.

Required: No

SupportedIdentityProviders (p. 121)

A list of provider names for the identity providers that are supported on this client.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 32.

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

Required: No

UserPoolId (p. 121)

The user pool ID for the user pool where you want to create a user pool client.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

WriteAttributes (p. 121)

The user pool attributes that the app client can write to.

If your app client allows users to sign in through an identity provider, this array must include all attributes that are mapped to identity provider attributes. Amazon Cognito updates mapped attributes when users sign in to your application through an identity provider. If your app client lacks write access to a mapped attribute, Amazon Cognito throws an error when it attempts to update the attribute. For more information, see Specifying Identity Provider Attribute Mappings for Your User Pool.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: No

Response Syntax

```
{
   "UserPoolClient": {
      "AllowedOAuthFlows": [ "string" ],
      "AllowedOAuthFlowsUserPoolClient": boolean,
      "AllowedOAuthScopes": [ "string" ],
      "AnalyticsConfiguration": {
         "ApplicationId": "string",
         "ExternalId": "string",
         "RoleArn": "string",
         "UserDataShared": boolean
      },
      "CallbackURLs": [ "string" ],
      "ClientId": "string",
      "ClientName": "string",
```

```
      "ClientSecret": "string",
      "CreationDate": number,
      "DefaultRedirectURI": "string",
      "ExplicitAuthFlows": [ "string" ],
      "LastModifiedDate": number,
      "LogoutURLs": [ "string" ],
      "ReadAttributes": [ "string" ],
      "RefreshTokenValidity": number,
      "SupportedIdentityProviders": [ "string" ],
      "UserPoolId": "string",
      "WriteAttributes": [ "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.

UserPoolClient (p. 124)

The user pool client that was just created.

Type: UserPoolClientType (p. 376) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidOAuthFlowException

This exception is thrown when the specified OAuth flow is invalid.

HTTP Status Code: 400

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

LimitExceededException

This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

ScopeDoesNotExistException

This exception is thrown when the specified scope does not exist.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

CreateUserPoolDomain

Creates a new domain for a user pool.

Request Syntax

```
{
   "CustomDomainConfig": {
      "CertificateArn": "string"
   },
   "Domain": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

CustomDomainConfig (p. 127)

The configuration for a custom domain that hosts the sign-up and sign-in webpages for your application.

Provide this parameter only if you want to use a custom domain for your user pool. Otherwise, you can exclude this parameter and use the Amazon Cognito hosted domain instead.

For more information about the hosted domain and custom domains, see Configuring a User Pool Domain.

Type: CustomDomainConfigType (p. 325) object

Required: No

Domain (p. 127)

The domain string.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 63.

Pattern: `^[a-z0-9](?:[a-z0-9\-]{0,61}[a-z0-9])?$`

Required: Yes

UserPoolId (p. 127)

The user pool ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "CloudFrontDomain": "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.

CloudFrontDomain (p. 128)

The Amazon CloudFront endpoint that you use as the target of the alias that you set up with your Domain Name Service (DNS) provider.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 63.

Pattern: `^[a-z0-9](?:[a-z0-9\-]{0,61}[a-z0-9])?$`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

LimitExceededException

This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

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

DeleteGroup

Deletes a group. Currently only groups with no members can be deleted.

Requires developer credentials.

Request Syntax

```
{
   "GroupName": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

GroupName (p. 130)

The name of the group.

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

UserPoolId (p. 130)

The user pool ID for the user pool.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

DeleteIdentityProvider

Deletes an identity provider for a user pool.

Request Syntax

```
{
   "ProviderName": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

ProviderName (p. 132)

The identity provider name.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 32.

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

Required: Yes

UserPoolId (p. 132)

The user pool ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UnsupportedIdentityProviderException

This exception is thrown when the specified identifier is not supported.

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

DeleteResourceServer

Deletes a resource server.

Request Syntax

```
{
   "Identifier": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

Identifier (p. 134)

The identifier for the resource server.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `[\x21\x23-\x5B\x5D-\x7E]+`

Required: Yes

UserPoolId (p. 134)

The user pool ID for the user pool that hosts the resource server.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

DeleteUser

Allows a user to delete himself or herself.

Request Syntax

```
{
   "AccessToken": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AccessToken (p. 136)

The access token from a request to delete a user.

Type: String

Pattern: `[A-Za-z0-9-_=.]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user is not confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

DeleteUserAttributes

Deletes the attributes for a user.

Request Syntax

```
{
   "AccessToken": "string",
   "UserAttributeNames": [ "string" ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AccessToken (p. 138)

The access token used in the request to delete user attributes.

Type: String

Pattern: `[A-Za-z0-9-_=.]+`

Required: Yes

UserAttributeNames (p. 138)

An array of strings representing the user attribute names you wish to delete.

For custom attributes, you must prepend the `custom:` prefix to the attribute name.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 32.

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

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user is not confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

DeleteUserPool

Deletes the specified Amazon Cognito user pool.

Request Syntax

```
{
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

UserPoolId (p. 140)

The user pool ID for the user pool you want to delete.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserImportInProgressException

This exception is thrown when you are trying to modify a user pool while a user import job is in progress for that pool.

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

DeleteUserPoolClient

Allows the developer to delete the user pool client.

Request Syntax

```
{
   "ClientId": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

ClientId (p. 142)

The app client ID of the app associated with the user pool.

Type: String

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

Pattern: `[\w+]+`

Required: Yes

UserPoolId (p. 142)

The user pool ID for the user pool where you want to delete the client.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

DeleteUserPoolDomain

Deletes a domain for a user pool.

Request Syntax

```
{
   "Domain": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

Domain (p. 144)

The domain string.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 63.

Pattern: `^[a-z0-9](?:[a-z0-9\-]{0,61}[a-z0-9])?$`

Required: Yes

UserPoolId (p. 144)

The user pool ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

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

DescribeIdentityProvider

Gets information about a specific identity provider.

Request Syntax

```
{
   "ProviderName": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

ProviderName (p. 146)

The identity provider name.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 32.

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

Required: Yes

UserPoolId (p. 146)

The user pool ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "IdentityProvider": {
      "AttributeMapping": {
         "string" : "string"
      },
      "CreationDate": number,
      "IdpIdentifiers": [ "string" ],
      "LastModifiedDate": number,
      "ProviderDetails": {
         "string" : "string"
      },
      "ProviderName": "string",
      "ProviderType": "string",
```

```
        "UserPoolId": "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.

IdentityProvider (p. 146)

The identity provider that was deleted.

Type: IdentityProviderType (p. 338) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

DescribeResourceServer

Describes a resource server.

Request Syntax

```
{
   "Identifier": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

Identifier (p. 149)

The identifier for the resource server

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `[\x21\x23-\x5B\x5D-\x7E]+`

Required: Yes

UserPoolId (p. 149)

The user pool ID for the user pool that hosts the resource server.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "ResourceServer": {
      "Identifier": "string",
      "Name": "string",
      "Scopes": [
         {
            "ScopeDescription": "string",
            "ScopeName": "string"
         }
      ],
      "UserPoolId": "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.

ResourceServer (p. 149)

The resource server.

Type: ResourceServerType (p. 355) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

DescribeRiskConfiguration

Describes the risk configuration.

Request Syntax

```
{
   "ClientId": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

ClientId (p. 152)

The app client ID.

Type: String

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

Pattern: `[\w+]+`

Required: No

UserPoolId (p. 152)

The user pool ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "RiskConfiguration": {
      "AccountTakeoverRiskConfiguration": {
         "Actions": {
            "HighAction": {
               "EventAction": "string",
               "Notify": boolean
            },
            "LowAction": {
               "EventAction": "string",
               "Notify": boolean
            },
```

```
            "MediumAction": {
               "EventAction": "string",
               "Notify": boolean
            }
         },
         "NotifyConfiguration": {
            "BlockEmail": {
               "HtmlBody": "string",
               "Subject": "string",
               "TextBody": "string"
            },
            "From": "string",
            "MfaEmail": {
               "HtmlBody": "string",
               "Subject": "string",
               "TextBody": "string"
            },
            "NoActionEmail": {
               "HtmlBody": "string",
               "Subject": "string",
               "TextBody": "string"
            },
            "ReplyTo": "string",
            "SourceArn": "string"
         }
      },
      "ClientId": "string",
      "CompromisedCredentialsRiskConfiguration": {
         "Actions": {
            "EventAction": "string"
         },
         "EventFilter": [ "string" ]
      },
      "LastModifiedDate": number,
      "RiskExceptionConfiguration": {
         "BlockedIPRangeList": [ "string" ],
         "SkippedIPRangeList": [ "string" ]
      },
      "UserPoolId": "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.

RiskConfiguration (p. 152)

The risk configuration.

Type: RiskConfigurationType (p. 357) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserPoolAddOnNotEnabledException

This exception is thrown when user pool add-ons are not enabled.

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

DescribeUserImportJob

Describes the user import job.

Request Syntax

```
{
   "JobId": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

JobId (p. 155)

The job ID for the user import job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `import-[0-9a-zA-Z-]+`

Required: Yes

UserPoolId (p. 155)

The user pool ID for the user pool that the users are being imported into.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "UserImportJob": {
      "CloudWatchLogsRoleArn": "string",
      "CompletionDate": number,
      "CompletionMessage": "string",
      "CreationDate": number,
      "FailedUsers": number,
      "ImportedUsers": number,
      "JobId": "string",
      "JobName": "string",
      "PreSignedUrl": "string",
      "SkippedUsers": number,
      "StartDate": number,
```

```
        "Status": "string",
        "UserPoolId": "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.

UserImportJob (p. 155)

The job object that represents the user import job.

Type: UserImportJobType (p. 371) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

DescribeUserPool

Returns the configuration information and metadata of the specified user pool.

Request Syntax

```
{
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

UserPoolId (p. 158)

The user pool ID for the user pool you want to describe.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "UserPool": {
      "AdminCreateUserConfig": {
         "AllowAdminCreateUserOnly": boolean,
         "InviteMessageTemplate": {
            "EmailMessage": "string",
            "EmailSubject": "string",
            "SMSMessage": "string"
         },
         "UnusedAccountValidityDays": number
      },
      "AliasAttributes": [ "string" ],
      "Arn": "string",
      "AutoVerifiedAttributes": [ "string" ],
      "CreationDate": number,
      "CustomDomain": "string",
      "DeviceConfiguration": {
         "ChallengeRequiredOnNewDevice": boolean,
         "DeviceOnlyRememberedOnUserPrompt": boolean
      },
      "Domain": "string",
      "EmailConfiguration": {
         "ReplyToEmailAddress": "string",
         "SourceArn": "string"
      },
      "EmailConfigurationFailure": "string",
```

```
"EmailVerificationMessage": "string",
"EmailVerificationSubject": "string",
"EstimatedNumberOfUsers": number,
"Id": "string",
"LambdaConfig": {
   "CreateAuthChallenge": "string",
   "CustomMessage": "string",
   "DefineAuthChallenge": "string",
   "PostAuthentication": "string",
   "PostConfirmation": "string",
   "PreAuthentication": "string",
   "PreSignUp": "string",
   "PreTokenGeneration": "string",
   "UserMigration": "string",
   "VerifyAuthChallengeResponse": "string"
},
"LastModifiedDate": number,
"MfaConfiguration": "string",
"Name": "string",
"Policies": {
   "PasswordPolicy": {
      "MinimumLength": number,
      "RequireLowercase": boolean,
      "RequireNumbers": boolean,
      "RequireSymbols": boolean,
      "RequireUppercase": boolean
   }
},
"SchemaAttributes": [
   {
      "AttributeDataType": "string",
      "DeveloperOnlyAttribute": boolean,
      "Mutable": boolean,
      "Name": "string",
      "NumberAttributeConstraints": {
         "MaxValue": "string",
         "MinValue": "string"
      },
      "Required": boolean,
      "StringAttributeConstraints": {
         "MaxLength": "string",
         "MinLength": "string"
      }
   }
],
"SmsAuthenticationMessage": "string",
"SmsConfiguration": {
   "ExternalId": "string",
   "SnsCallerArn": "string"
},
"SmsConfigurationFailure": "string",
"SmsVerificationMessage": "string",
"Status": "string",
"UsernameAttributes": [ "string" ],
"UserPoolAddOns": {
   "AdvancedSecurityMode": "string"
},
"UserPoolTags": {
   "string" : "string"
},
"VerificationMessageTemplate": {
   "DefaultEmailOption": "string",
   "EmailMessage": "string",
   "EmailMessageByLink": "string",
   "EmailSubject": "string",
   "EmailSubjectByLink": "string",
```

```
        "SmsMessage": "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.

UserPool (p. 158)

The container of metadata returned by the server to describe the pool.

Type: UserPoolType (p. 383) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserPoolTaggingException

This exception is thrown when a user pool tag cannot be set or updated.

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

DescribeUserPoolClient

Client method for returning the configuration information and metadata of the specified user pool app client.

Request Syntax

```
{
   "ClientId": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

ClientId (p. 162)

The app client ID of the app associated with the user pool.

Type: String

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

Pattern: `[\w+]+`

Required: Yes

UserPoolId (p. 162)

The user pool ID for the user pool you want to describe.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "UserPoolClient": {
      "AllowedOAuthFlows": [ "string" ],
      "AllowedOAuthFlowsUserPoolClient": boolean,
      "AllowedOAuthScopes": [ "string" ],
      "AnalyticsConfiguration": {
         "ApplicationId": "string",
         "ExternalId": "string",
         "RoleArn": "string",
         "UserDataShared": boolean
      },
```

```
      "CallbackURLs": [ "string" ],
      "ClientId": "string",
      "ClientName": "string",
      "ClientSecret": "string",
      "CreationDate": number,
      "DefaultRedirectURI": "string",
      "ExplicitAuthFlows": [ "string" ],
      "LastModifiedDate": number,
      "LogoutURLs": [ "string" ],
      "ReadAttributes": [ "string" ],
      "RefreshTokenValidity": number,
      "SupportedIdentityProviders": [ "string" ],
      "UserPoolId": "string",
      "WriteAttributes": [ "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.

UserPoolClient (p. 162)

The user pool client from a server response to describe the user pool client.

Type: UserPoolClientType (p. 376) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

DescribeUserPoolDomain

Gets information about a domain.

Request Syntax

```
{
   "Domain": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

Domain (p. 165)

The domain string.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 63.

Pattern: `^[a-z0-9](?:[a-z0-9\-]{0,61}[a-z0-9])?$`

Required: Yes

Response Syntax

```
{
   "DomainDescription": {
      "AWSAccountId": "string",
      "CloudFrontDistribution": "string",
      "CustomDomainConfig": {
         "CertificateArn": "string"
      },
      "Domain": "string",
      "S3Bucket": "string",
      "Status": "string",
      "UserPoolId": "string",
      "Version": "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.

DomainDescription (p. 165)

A domain description object containing information about the domain.

Type: DomainDescriptionType (p. 329) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

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

ForgetDevice

Forgets the specified device.

Request Syntax

```
{
   "AccessToken": "string",
   "DeviceKey": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AccessToken (p. 167)

The access token for the forgotten device request.

Type: String

Pattern: `[A-Za-z0-9-_=.]+`

Required: No

DeviceKey (p. 167)

The device key.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-f-]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidUserPoolConfigurationException

This exception is thrown when the user pool configuration is invalid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user is not confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

ForgotPassword

Calling this API causes a message to be sent to the end user with a confirmation code that is required to change the user's password. For the `Username` parameter, you can use the username or user alias. If a verified phone number exists for the user, the confirmation code is sent to the phone number. Otherwise, if a verified email exists, the confirmation code is sent to the email. If neither a verified phone number nor a verified email exists, `InvalidParameterException` is thrown. To use the confirmation code for resetting the password, call ConfirmForgotPassword (p. 93).

Request Syntax

```
{
   "AnalyticsMetadata": {
      "AnalyticsEndpointId": "string"
   },
   "ClientId": "string",
   "SecretHash": "string",
   "UserContextData": {
      "EncodedData": "string"
   },
   "Username": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AnalyticsMetadata (p. 169)

The Amazon Pinpoint analytics metadata for collecting metrics for `ForgotPassword` calls.

Type: AnalyticsMetadataType (p. 314) object

Required: No

ClientId (p. 169)

The ID of the client associated with the user pool.

Type: String

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

Pattern: `[\w+]+`

Required: Yes

SecretHash (p. 169)

A keyed-hash message authentication code (HMAC) calculated using the secret key of a user pool client and username plus the client ID in the message.

Type: String

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

Pattern: `[\w+=/]+`

Required: No

UserContextData (p. 169)

Contextual data such as the user's device fingerprint, IP address, or location used for evaluating the risk of an unexpected event by Amazon Cognito advanced security.

Type: UserContextDataType (p. 370) object

Required: No

Username (p. 169)

The user name of the user for whom you want to enter a code to reset a forgotten password.

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

Response Syntax

```
{
   "CodeDeliveryDetails": {
      "AttributeName": "string",
      "DeliveryMedium": "string",
      "Destination": "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.

CodeDeliveryDetails (p. 170)

The code delivery details returned by the server in response to the request to reset a password.

Type: CodeDeliveryDetailsType (p. 321) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

CodeDeliveryFailureException

This exception is thrown when a verification code fails to deliver successfully.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidEmailRoleAccessPolicyException

This exception is thrown when Amazon Cognito is not allowed to use your email identity. HTTP status code: 400.

HTTP Status Code: 400

InvalidLambdaResponseException

This exception is thrown when the Amazon Cognito service encounters an invalid AWS Lambda response.

HTTP Status Code: 400

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidSmsRoleAccessPolicyException

This exception is returned when the role provided for SMS configuration does not have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is invalid for the role provided for SMS configuration. This can happen if you do not trust **cognito-idp.amazonaws.com** or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.

HTTP Status Code: 400

LimitExceededException

This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UnexpectedLambdaException

This exception is thrown when the Amazon Cognito service encounters an unexpected exception with the AWS Lambda service.

HTTP Status Code: 400

UserLambdaValidationException

This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user is not confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

GetCSVHeader

Gets the header information for the .csv file to be used as input for the user import job.

Request Syntax

```
{
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

UserPoolId (p. 173)

The user pool ID for the user pool that the users are to be imported into.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "CSVHeader": [ "string" ],
   "UserPoolId": "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.

CSVHeader (p. 173)

The header information for the .csv file for the user import job.

Type: Array of strings

UserPoolId (p. 173)

The user pool ID for the user pool that the users are to be imported into.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

GetDevice

Gets the device.

Request Syntax

```
{
   "AccessToken": "string",
   "DeviceKey": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AccessToken (p. 175)

The access token.

Type: String

Pattern: `[A-Za-z0-9-_=.]+`

Required: No

DeviceKey (p. 175)

The device key.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-f-]+`

Required: Yes

Response Syntax

```
{
   "Device": {
      "DeviceAttributes": [
         {
            "Name": "string",
            "Value": "string"
         }
      ],
      "DeviceCreateDate": number,
      "DeviceKey": "string",
      "DeviceLastAuthenticatedDate": number,
      "DeviceLastModifiedDate": number
   }
}
```

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.

Device (p. 175)

The device.

Type: DeviceType (p. 328) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidUserPoolConfigurationException

This exception is thrown when the user pool configuration is invalid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user is not confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

GetGroup

Gets a group.

Requires developer credentials.

Request Syntax

```
{
   "GroupName": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

GroupName (p. 178)

The name of the group.

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

UserPoolId (p. 178)

The user pool ID for the user pool.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "Group": {
      "CreationDate": number,
      "Description": "string",
      "GroupName": "string",
      "LastModifiedDate": number,
      "Precedence": number,
      "RoleArn": "string",
      "UserPoolId": "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.

Group (p. 178)

The group object for the group.

Type: GroupType (p. 335) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

GetIdentityProviderByIdentifier

Gets the specified identity provider.

Request Syntax

```
{
   "IdpIdentifier": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

IdpIdentifier (p. 181)

The identity provider ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 40.

Pattern: `[\w\s+=.@-]+`

Required: Yes

UserPoolId (p. 181)

The user pool ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "IdentityProvider": {
      "AttributeMapping": {
         "string" : "string"
      },
      "CreationDate": number,
      "IdpIdentifiers": [ "string" ],
      "LastModifiedDate": number,
      "ProviderDetails": {
         "string" : "string"
      },
      "ProviderName": "string",
      "ProviderType": "string",
```

```
        "UserPoolId": "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.

IdentityProvider (p. 181)

The identity provider object.

Type: IdentityProviderType (p. 338) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

GetSigningCertificate

This method takes a user pool ID, and returns the signing certificate.

Request Syntax

```
{
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

UserPoolId (p. 184)

The user pool ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "Certificate": "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.

Certificate (p. 184)

The signing certificate.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

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

GetUICustomization

Gets the UI Customization information for a particular app client's app UI, if there is something set. If nothing is set for the particular client, but there is an existing pool level customization (app `clientId` will be `ALL`), then that is returned. If nothing is present, then an empty shape is returned.

Request Syntax

```
{
   "ClientId": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

ClientId (p. 186)

The client ID for the client app.

Type: String

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

Pattern: `[\w+]+`

Required: No

UserPoolId (p. 186)

The user pool ID for the user pool.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "UICustomization": {
      "ClientId": "string",
      "CreationDate": number,
      "CSS": "string",
      "CSSVersion": "string",
      "ImageUrl": "string",
      "LastModifiedDate": number,
      "UserPoolId": "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.

UICustomization (p. 186)

The UI customization information.

Type: UICustomizationType (p. 368) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

GetUser

Gets the user attributes and metadata for a user.

Request Syntax

```
{
   "AccessToken": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AccessToken (p. 189)

The access token returned by the server response to get information about the user.

Type: String

Pattern: `[A-Za-z0-9-_=.]+`

Required: Yes

Response Syntax

```
{
   "MFAOptions": [
      {
         "AttributeName": "string",
         "DeliveryMedium": "string"
      }
   ],
   "PreferredMfaSetting": "string",
   "UserAttributes": [
      {
         "Name": "string",
         "Value": "string"
      }
   ],
   "UserMFASettingList": [ "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.

MFAOptions (p. 189)

Specifies the options for MFA (e.g., email or phone number).

Type: Array of MFAOptionType (p. 344) objects

PreferredMfaSetting (p. 189)

The user's preferred MFA setting.

Type: String

UserAttributes (p. 189)

An array of name-value pairs representing user attributes.

For custom attributes, you must prepend the `custom:` prefix to the attribute name.

Type: Array of AttributeType (p. 315) objects

UserMFASettingList (p. 189)

The list of the user's MFA settings.

Type: Array of strings

Username (p. 189)

The user name of the user you wish to retrieve from the get user request.

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. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user is not confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

GetUserAttributeVerificationCode

Gets the user attribute verification code for the specified attribute name.

Request Syntax

```
{
   "AccessToken": "string",
   "AttributeName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AccessToken (p. 192)

The access token returned by the server response to get the user attribute verification code.

Type: String

Pattern: `[A-Za-z0-9-_=.]+`

Required: Yes

AttributeName (p. 192)

The attribute name returned by the server response to get the user attribute verification code.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 32.

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

Required: Yes

Response Syntax

```
{
   "CodeDeliveryDetails": {
      "AttributeName": "string",
      "DeliveryMedium": "string",
      "Destination": "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.

CodeDeliveryDetails (p. 192)

The code delivery details returned by the server in response to the request to get the user attribute verification code.

Type: CodeDeliveryDetailsType (p. 321) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

CodeDeliveryFailureException

This exception is thrown when a verification code fails to deliver successfully.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidEmailRoleAccessPolicyException

This exception is thrown when Amazon Cognito is not allowed to use your email identity. HTTP status code: 400.

HTTP Status Code: 400

InvalidLambdaResponseException

This exception is thrown when the Amazon Cognito service encounters an invalid AWS Lambda response.

HTTP Status Code: 400

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidSmsRoleAccessPolicyException

This exception is returned when the role provided for SMS configuration does not have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is invalid for the role provided for SMS configuration. This can happen if you do not trust **cognito-idp.amazonaws.com** or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.

HTTP Status Code: 400

LimitExceededException

This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UnexpectedLambdaException

This exception is thrown when the Amazon Cognito service encounters an unexpected exception with the AWS Lambda service.

HTTP Status Code: 400

UserLambdaValidationException

This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user is not confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

GetUserPoolMfaConfig

Gets the user pool multi-factor authentication (MFA) configuration.

Request Syntax

```
{
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

UserPoolId (p. 196)

The user pool ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "MfaConfiguration": "string",
   "SmsMfaConfiguration": {
      "SmsAuthenticationMessage": "string",
      "SmsConfiguration": {
         "ExternalId": "string",
         "SnsCallerArn": "string"
      }
   },
   "SoftwareTokenMfaConfiguration": {
      "Enabled": boolean
   }
}
```

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.

MfaConfiguration (p. 196)

The multi-factor (MFA) configuration.

Type: String

Valid Values: `OFF | ON | OPTIONAL`

SmsMfaConfiguration (p. 196)

The SMS text message multi-factor (MFA) configuration.

Type: SmsMfaConfigType (p. 363) object

SoftwareTokenMfaConfiguration (p. 196)

The software token multi-factor (MFA) configuration.

Type: SoftwareTokenMfaConfigType (p. 365) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

GlobalSignOut

Signs out users from all devices.

Request Syntax

```
{
   "AccessToken": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AccessToken (p. 199)

The access token.

Type: String

Pattern: `[A-Za-z0-9-_=.]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user is not confirmed successfully.

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

InitiateAuth

Initiates the authentication flow.

Request Syntax

```
{
   "AnalyticsMetadata": {
      "AnalyticsEndpointId": "string"
   },
   "AuthFlow": "string",
   "AuthParameters": {
      "string" : "string"
   },
   "ClientId": "string",
   "ClientMetadata": {
      "string" : "string"
   },
   "UserContextData": {
      "EncodedData": "string"
   }
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AnalyticsMetadata (p. 201)

The Amazon Pinpoint analytics metadata for collecting metrics for `InitiateAuth` calls.

Type: AnalyticsMetadataType (p. 314) object

Required: No

AuthFlow (p. 201)

The authentication flow for this call to execute. The API action will depend on this value. For example:

- `REFRESH_TOKEN_AUTH` will take in a valid refresh token and return new tokens.
- `USER_SRP_AUTH` will take in `USERNAME` and `SRP_A` and return the SRP variables to be used for next challenge execution.
- `USER_PASSWORD_AUTH` will take in `USERNAME` and `PASSWORD` and return the next challenge or tokens.

Valid values include:

- `USER_SRP_AUTH`: Authentication flow for the Secure Remote Password (SRP) protocol.
- `REFRESH_TOKEN_AUTH/REFRESH_TOKEN`: Authentication flow for refreshing the access token and ID token by supplying a valid refresh token.
- `CUSTOM_AUTH`: Custom authentication flow.
- `USER_PASSWORD_AUTH`: Non-SRP authentication flow; USERNAME and PASSWORD are passed directly. If a user migration Lambda trigger is set, this flow will invoke the user migration Lambda if the USERNAME is not found in the user pool.

`ADMIN_NO_SRP_AUTH` is not a valid value.

Type: String

Valid Values: `USER_SRP_AUTH` | `REFRESH_TOKEN_AUTH` | `REFRESH_TOKEN` | `CUSTOM_AUTH` | `ADMIN_NO_SRP_AUTH` | `USER_PASSWORD_AUTH`

Required: Yes

AuthParameters (p. 201)

The authentication parameters. These are inputs corresponding to the `AuthFlow` that you are invoking. The required values depend on the value of `AuthFlow`:

- For `USER_SRP_AUTH`: `USERNAME` (required), `SRP_A` (required), `SECRET_HASH` (required if the app client is configured with a client secret), `DEVICE_KEY`
- For `REFRESH_TOKEN_AUTH/REFRESH_TOKEN`: `REFRESH_TOKEN` (required), `SECRET_HASH` (required if the app client is configured with a client secret), `DEVICE_KEY`
- For `CUSTOM_AUTH`: `USERNAME` (required), `SECRET_HASH` (if app client is configured with client secret), `DEVICE_KEY`

Type: String to string map

Required: No

ClientId (p. 201)

The app client ID.

Type: String

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

Pattern: `[\w+]+`

Required: Yes

ClientMetadata (p. 201)

This is a random key-value pair map which can contain any key and will be passed to your PreAuthentication Lambda trigger as-is. It can be used to implement additional validations around authentication.

Type: String to string map

Required: No

UserContextData (p. 201)

Contextual data such as the user's device fingerprint, IP address, or location used for evaluating the risk of an unexpected event by Amazon Cognito advanced security.

Type: UserContextDataType (p. 370) object

Required: No

Response Syntax

```
{
   "AuthenticationResult": {
      "AccessToken": "string",
      "ExpiresIn": number,
```

```
      "IdToken": "string",
      "NewDeviceMetadata": {
         "DeviceGroupKey": "string",
         "DeviceKey": "string"
      },
      "RefreshToken": "string",
      "TokenType": "string"
   },
   "ChallengeName": "string",
   "ChallengeParameters": {
      "string" : "string"
   },
   "Session": "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.

AuthenticationResult (p. 202)

The result of the authentication response. This is only returned if the caller does not need to pass another challenge. If the caller does need to pass another challenge before it gets tokens, `ChallengeName`, `ChallengeParameters`, and `Session` are returned.

Type: AuthenticationResultType (p. 316) object

ChallengeName (p. 202)

The name of the challenge which you are responding to with this call. This is returned to you in the `AdminInitiateAuth` response if you need to pass another challenge.

Valid values include the following. Note that all of these challenges require `USERNAME` and `SECRET_HASH` (if applicable) in the parameters.

- `SMS_MFA`: Next challenge is to supply an `SMS_MFA_CODE`, delivered via SMS.
- `PASSWORD_VERIFIER`: Next challenge is to supply `PASSWORD_CLAIM_SIGNATURE`, `PASSWORD_CLAIM_SECRET_BLOCK`, and `TIMESTAMP` after the client-side SRP calculations.
- `CUSTOM_CHALLENGE`: This is returned if your custom authentication flow determines that the user should pass another challenge before tokens are issued.
- `DEVICE_SRP_AUTH`: If device tracking was enabled on your user pool and the previous challenges were passed, this challenge is returned so that Amazon Cognito can start tracking this device.
- `DEVICE_PASSWORD_VERIFIER`: Similar to `PASSWORD_VERIFIER`, but for devices only.
- `NEW_PASSWORD_REQUIRED`: For users which are required to change their passwords after successful first login. This challenge should be passed with `NEW_PASSWORD` and any other required attributes.

Type: String

Valid Values: `SMS_MFA | SOFTWARE_TOKEN_MFA | SELECT_MFA_TYPE | MFA_SETUP | PASSWORD_VERIFIER | CUSTOM_CHALLENGE | DEVICE_SRP_AUTH | DEVICE_PASSWORD_VERIFIER | ADMIN_NO_SRP_AUTH | NEW_PASSWORD_REQUIRED`

ChallengeParameters (p. 202)

The challenge parameters. These are returned to you in the `InitiateAuth` response if you need to pass another challenge. The responses in this parameter should be used to compute inputs to the next call (`RespondToAuthChallenge`).

All challenges require `USERNAME` and `SECRET_HASH` (if applicable).

Type: String to string map

Session (p. 202)

The session which should be passed both ways in challenge-response calls to the service. If the InitiateAuth (p. 201) or RespondToAuthChallenge (p. 238) API call determines that the caller needs to go through another challenge, they return a session with other challenge parameters. This session should be passed as it is to the next `RespondToAuthChallenge` API call.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidLambdaResponseException

This exception is thrown when the Amazon Cognito service encounters an invalid AWS Lambda response.

HTTP Status Code: 400

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidUserPoolConfigurationException

This exception is thrown when the user pool configuration is invalid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UnexpectedLambdaException

This exception is thrown when the Amazon Cognito service encounters an unexpected exception with the AWS Lambda service.

HTTP Status Code: 400

UserLambdaValidationException

This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user is not confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

ListDevices

Lists the devices.

Request Syntax

```
{
   "AccessToken": "string",
   "Limit": number,
   "PaginationToken": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AccessToken (p. 206)

The access tokens for the request to list devices.

Type: String

Pattern: `[A-Za-z0-9-_=.]+`

Required: Yes

Limit (p. 206)

The limit of the device request.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 60.

Required: No

PaginationToken (p. 206)

The pagination token for the list request.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

Required: No

Response Syntax

```
{
   "Devices": [
      {
         "DeviceAttributes": [
            {
```

```
                "Name": "string",
                "Value": "string"
            }
        ],
        "DeviceCreateDate": number,
        "DeviceKey": "string",
        "DeviceLastAuthenticatedDate": number,
        "DeviceLastModifiedDate": number
    }
  ],
  "PaginationToken": "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.

Devices (p. 206)

The devices returned in the list devices response.

Type: Array of DeviceType (p. 328) objects

PaginationToken (p. 206)

The pagination token for the list device response.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidUserPoolConfigurationException

This exception is thrown when the user pool configuration is invalid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user is not confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

ListGroups

Lists the groups associated with a user pool.

Requires developer credentials.

Request Syntax

```
{
   "Limit": number,
   "NextToken": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

Limit (p. 209)

The limit of the request to list groups.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 60.

Required: No

NextToken (p. 209)

An identifier that was returned from the previous call to this operation, which can be used to return the next set of items in the list.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

Required: No

UserPoolId (p. 209)

The user pool ID for the user pool.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
```

```
   "Groups": [
      {
         "CreationDate": number,
         "Description": "string",
         "GroupName": "string",
         "LastModifiedDate": number,
         "Precedence": number,
         "RoleArn": "string",
         "UserPoolId": "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. 209)

The group objects for the groups.

Type: Array of GroupType (p. 335) objects

NextToken (p. 209)

An identifier that was returned from the previous call to this operation, which can be used to return the next set of items in the list.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

ListIdentityProviders

Lists information about all identity providers for a user pool.

Request Syntax

```
{
   "MaxResults": number,
   "NextToken": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

MaxResults (p. 212)

The maximum number of identity providers to return.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 60.

Required: No

NextToken (p. 212)

A pagination token.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

Required: No

UserPoolId (p. 212)

The user pool ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "NextToken": "string",
```

```
   "Providers": [
      {
         "CreationDate": number,
         "LastModifiedDate": number,
         "ProviderName": "string",
         "ProviderType": "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. 212)

A pagination token.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

Providers (p. 212)

A list of identity provider objects.

Type: Array of ProviderDescription (p. 352) objects

Array Members: Minimum number of 0 items. Maximum number of 50 items.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

ListResourceServers

Lists the resource servers for a user pool.

Request Syntax

```
{
   "MaxResults": number,
   "NextToken": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

MaxResults (p. 215)

The maximum number of resource servers to return.

Type: Integer

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

Required: No

NextToken (p. 215)

A pagination token.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

Required: No

UserPoolId (p. 215)

The user pool ID for the user pool.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "NextToken": "string",
```

```
   "ResourceServers": [
      {
         "Identifier": "string",
         "Name": "string",
         "Scopes": [
            {
               "ScopeDescription": "string",
               "ScopeName": "string"
            }
         ],
         "UserPoolId": "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. 215)

A pagination token.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

ResourceServers (p. 215)

The resource servers.

Type: Array of ResourceServerType (p. 355) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

ListUserImportJobs

Lists the user import jobs.

Request Syntax

```
{
   "MaxResults": number,
   "PaginationToken": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

MaxResults (p. 218)

The maximum number of import jobs you want the request to return.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 60.

Required: Yes

PaginationToken (p. 218)

An identifier that was returned from the previous call to `ListUserImportJobs`, which can be used to return the next set of import jobs in the list.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

Required: No

UserPoolId (p. 218)

The user pool ID for the user pool that the users are being imported into.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "PaginationToken": "string",
```

```
   "UserImportJobs": [
      {
         "CloudWatchLogsRoleArn": "string",
         "CompletionDate": number,
         "CompletionMessage": "string",
         "CreationDate": number,
         "FailedUsers": number,
         "ImportedUsers": number,
         "JobId": "string",
         "JobName": "string",
         "PreSignedUrl": "string",
         "SkippedUsers": number,
         "StartDate": number,
         "Status": "string",
         "UserPoolId": "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.

PaginationToken (p. 218)

An identifier that can be used to return the next set of user import jobs in the list.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

UserImportJobs (p. 218)

The user import jobs.

Type: Array of UserImportJobType (p. 371) objects

Array Members: Minimum number of 1 item. Maximum number of 50 items.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

ListUserPoolClients

Lists the clients that have been created for the specified user pool.

Request Syntax

```
{
   "MaxResults": number,
   "NextToken": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

MaxResults (p. 221)

The maximum number of results you want the request to return when listing the user pool clients.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 60.

Required: No

NextToken (p. 221)

An identifier that was returned from the previous call to this operation, which can be used to return the next set of items in the list.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

Required: No

UserPoolId (p. 221)

The user pool ID for the user pool where you want to list user pool clients.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "NextToken": "string",
```

```
   "UserPoolClients": [
      {
         "ClientId": "string",
         "ClientName": "string",
         "UserPoolId": "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. 221)

An identifier that was returned from the previous call to this operation, which can be used to return the next set of items in the list.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

UserPoolClients (p. 221)

The user pool clients in the response that lists user pool clients.

Type: Array of UserPoolClientDescription (p. 375) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

ListUserPools

Lists the user pools associated with an AWS account.

Request Syntax

```
{
   "MaxResults": number,
   "NextToken": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

MaxResults (p. 224)

The maximum number of results you want the request to return when listing the user pools.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 60.

Required: Yes

NextToken (p. 224)

An identifier that was returned from the previous call to this operation, which can be used to return the next set of items in the list.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

Required: No

Response Syntax

```
{
   "NextToken": "string",
   "UserPools": [
      {
         "CreationDate": number,
         "Id": "string",
         "LambdaConfig": {
            "CreateAuthChallenge": "string",
            "CustomMessage": "string",
            "DefineAuthChallenge": "string",
            "PostAuthentication": "string",
            "PostConfirmation": "string",
            "PreAuthentication": "string",
            "PreSignUp": "string",
```

```
            "PreTokenGeneration": "string",
            "UserMigration": "string",
            "VerifyAuthChallengeResponse": "string"
         },
         "LastModifiedDate": number,
         "Name": "string",
         "Status": "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. 224)

An identifier that was returned from the previous call to this operation, which can be used to return the next set of items in the list.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

UserPools (p. 224)

The user pools from the response to list users.

Type: Array of UserPoolDescriptionType (p. 380) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

ListUsers

Lists the users in the Amazon Cognito user pool.

Request Syntax

```
{
   "AttributesToGet": [ "string" ],
   "Filter": "string",
   "Limit": number,
   "PaginationToken": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AttributesToGet (p. 227)

An array of strings, where each string is the name of a user attribute to be returned for each user in the search results. If the array is null, all attributes are returned.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 32.

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

Required: No

Filter (p. 227)

A filter string of the form "*AttributeName Filter-Type* "*AttributeValue*"". Quotation marks within the filter string must be escaped using the backslash (\) character. For example, "`family_name` = \"Reddy\"".

- *AttributeName*: The name of the attribute to search for. You can only search for one attribute at a time.
- *Filter-Type*: For an exact match, use =, for example, "`given_name` = \"Jon\"". For a prefix ("starts with") match, use ^=, for example, "`given_name` ^= \"Jon\"".
- *AttributeValue*: The attribute value that must be matched for each user.

If the filter string is empty, `ListUsers` returns all users in the user pool.

You can only search for the following standard attributes:

- `username` (case-sensitive)
- `email`
- `phone_number`
- `name`
- `given_name`
- `family_name`

- `preferred_username`
- `cognito:user_status` (called **Status** in the Console) (case-insensitive)
- `status (called **Enabled** in the Console) (case-sensitive)`
- `sub`

Custom attributes are not searchable.

For more information, see Searching for Users Using the ListUsers API and Examples of Using the ListUsers API in the *Amazon Cognito Developer Guide*.

Type: String

Length Constraints: Maximum length of 256.

Required: No

Limit (p. 227)

Maximum number of users to be returned.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 60.

Required: No

PaginationToken (p. 227)

An identifier that was returned from the previous call to this operation, which can be used to return the next set of items in the list.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

Required: No

UserPoolId (p. 227)

The user pool ID for the user pool on which the search should be performed.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "PaginationToken": "string",
   "Users": [
      {
         "Attributes": [
            {
               "Name": "string",
               "Value": "string"
```

```
            }
         ],
         "Enabled": boolean,
         "MFAOptions": [
            {
               "AttributeName": "string",
               "DeliveryMedium": "string"
            }
         ],
         "UserCreateDate": number,
         "UserLastModifiedDate": number,
         "Username": "string",
         "UserStatus": "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.

PaginationToken (p. 228)

An identifier that was returned from the previous call to this operation, which can be used to return the next set of items in the list.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

Users (p. 228)

The users returned in the request to list users.

Type: Array of UserType (p. 388) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

ListUsersInGroup

Lists the users in the specified group.

Requires developer credentials.

Request Syntax

```
{
   "GroupName": "string",
   "Limit": number,
   "NextToken": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

GroupName (p. 231)

The name of the group.

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

Limit (p. 231)

The limit of the request to list users.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 60.

Required: No

NextToken (p. 231)

An identifier that was returned from the previous call to this operation, which can be used to return the next set of items in the list.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

Required: No

UserPoolId (p. 231)

The user pool ID for the user pool.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "NextToken": "string",
   "Users": [
      {
         "Attributes": [
            {
               "Name": "string",
               "Value": "string"
            }
         ],
         "Enabled": boolean,
         "MFAOptions": [
            {
               "AttributeName": "string",
               "DeliveryMedium": "string"
            }
         ],
         "UserCreateDate": number,
         "UserLastModifiedDate": number,
         "Username": "string",
         "UserStatus": "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. 232)

An identifier that was returned from the previous call to this operation, which can be used to return the next set of items in the list.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

Users (p. 232)

The users returned in the request to list users.

Type: Array of UserType (p. 388) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

ResendConfirmationCode

Resends the confirmation (for confirmation of registration) to a specific user in the user pool.

Request Syntax

```
{
   "AnalyticsMetadata": {
      "AnalyticsEndpointId": "string"
   },
   "ClientId": "string",
   "SecretHash": "string",
   "UserContextData": {
      "EncodedData": "string"
   },
   "Username": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AnalyticsMetadata (p. 234)

The Amazon Pinpoint analytics metadata for collecting metrics for `ResendConfirmationCode` calls.

Type: AnalyticsMetadataType (p. 314) object

Required: No

ClientId (p. 234)

The ID of the client associated with the user pool.

Type: String

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

Pattern: `[\w+]+`

Required: Yes

SecretHash (p. 234)

A keyed-hash message authentication code (HMAC) calculated using the secret key of a user pool client and username plus the client ID in the message.

Type: String

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

Pattern: `[\w+=/]+`

Required: No

UserContextData (p. 234)

Contextual data such as the user's device fingerprint, IP address, or location used for evaluating the risk of an unexpected event by Amazon Cognito advanced security.

Type: UserContextDataType (p. 370) object

Required: No

Username (p. 234)

The user name of the user to whom you wish to resend a confirmation code.

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

Response Syntax

```
{
   "CodeDeliveryDetails": {
      "AttributeName": "string",
      "DeliveryMedium": "string",
      "Destination": "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.

CodeDeliveryDetails (p. 235)

The code delivery details returned by the server in response to the request to resend the confirmation code.

Type: CodeDeliveryDetailsType (p. 321) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

CodeDeliveryFailureException

This exception is thrown when a verification code fails to deliver successfully.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidEmailRoleAccessPolicyException

This exception is thrown when Amazon Cognito is not allowed to use your email identity. HTTP status code: 400.

HTTP Status Code: 400

InvalidLambdaResponseException

This exception is thrown when the Amazon Cognito service encounters an invalid AWS Lambda response.

HTTP Status Code: 400

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidSmsRoleAccessPolicyException

This exception is returned when the role provided for SMS configuration does not have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is invalid for the role provided for SMS configuration. This can happen if you do not trust **cognito-idp.amazonaws.com** or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.

HTTP Status Code: 400

LimitExceededException

This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UnexpectedLambdaException

This exception is thrown when the Amazon Cognito service encounters an unexpected exception with the AWS Lambda service.

HTTP Status Code: 400

UserLambdaValidationException

This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

RespondToAuthChallenge

Responds to the authentication challenge.

Request Syntax

```
{
   "AnalyticsMetadata": {
      "AnalyticsEndpointId": "string"
   },
   "ChallengeName": "string",
   "ChallengeResponses": {
      "string" : "string"
   },
   "ClientId": "string",
   "Session": "string",
   "UserContextData": {
      "EncodedData": "string"
   }
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AnalyticsMetadata (p. 238)

The Amazon Pinpoint analytics metadata for collecting metrics for `RespondToAuthChallenge` calls.

Type: AnalyticsMetadataType (p. 314) object

Required: No

ChallengeName (p. 238)

The challenge name. For more information, see InitiateAuth (p. 201).

`ADMIN_NO_SRP_AUTH` is not a valid value.

Type: String

Valid Values: `SMS_MFA | SOFTWARE_TOKEN_MFA | SELECT_MFA_TYPE | MFA_SETUP | PASSWORD_VERIFIER | CUSTOM_CHALLENGE | DEVICE_SRP_AUTH | DEVICE_PASSWORD_VERIFIER | ADMIN_NO_SRP_AUTH | NEW_PASSWORD_REQUIRED`

Required: Yes

ChallengeResponses (p. 238)

The challenge responses. These are inputs corresponding to the value of `ChallengeName`, for example:

- `SMS_MFA`: `SMS_MFA_CODE`, `USERNAME`, `SECRET_HASH` (if app client is configured with client secret).
- `PASSWORD_VERIFIER`: `PASSWORD_CLAIM_SIGNATURE`, `PASSWORD_CLAIM_SECRET_BLOCK`, `TIMESTAMP`, `USERNAME`, `SECRET_HASH` (if app client is configured with client secret).

- `NEW_PASSWORD_REQUIRED`: `NEW_PASSWORD`, any other required attributes, `USERNAME`, `SECRET_HASH` (if app client is configured with client secret).

Type: String to string map

Required: No

ClientId (p. 238)

The app client ID.

Type: String

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

Pattern: `[\w+]+`

Required: Yes

Session (p. 238)

The session which should be passed both ways in challenge-response calls to the service. If `InitiateAuth` or `RespondToAuthChallenge` API call determines that the caller needs to go through another challenge, they return a session with other challenge parameters. This session should be passed as it is to the next `RespondToAuthChallenge` API call.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Required: No

UserContextData (p. 238)

Contextual data such as the user's device fingerprint, IP address, or location used for evaluating the risk of an unexpected event by Amazon Cognito advanced security.

Type: UserContextDataType (p. 370) object

Required: No

Response Syntax

```
{
   "AuthenticationResult": {
      "AccessToken": "string",
      "ExpiresIn": number,
      "IdToken": "string",
      "NewDeviceMetadata": {
         "DeviceGroupKey": "string",
         "DeviceKey": "string"
      },
      "RefreshToken": "string",
      "TokenType": "string"
   },
   "ChallengeName": "string",
   "ChallengeParameters": {
      "string" : "string"
   },
   "Session": "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.

AuthenticationResult (p. 239)

The result returned by the server in response to the request to respond to the authentication challenge.

Type: AuthenticationResultType (p. 316) object

ChallengeName (p. 239)

The challenge name. For more information, see InitiateAuth (p. 201).

Type: String

Valid Values: `SMS_MFA | SOFTWARE_TOKEN_MFA | SELECT_MFA_TYPE | MFA_SETUP | PASSWORD_VERIFIER | CUSTOM_CHALLENGE | DEVICE_SRP_AUTH | DEVICE_PASSWORD_VERIFIER | ADMIN_NO_SRP_AUTH | NEW_PASSWORD_REQUIRED`

ChallengeParameters (p. 239)

The challenge parameters. For more information, see InitiateAuth (p. 201).

Type: String to string map

Session (p. 239)

The session which should be passed both ways in challenge-response calls to the service. If the InitiateAuth (p. 201) or RespondToAuthChallenge (p. 238) API call determines that the caller needs to go through another challenge, they return a session with other challenge parameters. This session should be passed as it is to the next `RespondToAuthChallenge` API call.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

AliasExistsException

This exception is thrown when a user tries to confirm the account with an email or phone number that has already been supplied as an alias from a different account. This exception tells user that an account with this email or phone already exists.

HTTP Status Code: 400

CodeMismatchException

This exception is thrown if the provided code does not match what the server was expecting.

HTTP Status Code: 400

ExpiredCodeException

This exception is thrown if a code has expired.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidLambdaResponseException

This exception is thrown when the Amazon Cognito service encounters an invalid AWS Lambda response.

HTTP Status Code: 400

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidPasswordException

This exception is thrown when the Amazon Cognito service encounters an invalid password.

HTTP Status Code: 400

InvalidSmsRoleAccessPolicyException

This exception is returned when the role provided for SMS configuration does not have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is invalid for the role provided for SMS configuration. This can happen if you do not trust **cognito-idp.amazonaws.com** or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.

HTTP Status Code: 400

InvalidUserPoolConfigurationException

This exception is thrown when the user pool configuration is invalid.

HTTP Status Code: 400

MFAMethodNotFoundException

This exception is thrown when Amazon Cognito cannot find a multi-factor authentication (MFA) method.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

SoftwareTokenMFANotFoundException

This exception is thrown when the software token TOTP multi-factor authentication (MFA) is not enabled for the user pool.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UnexpectedLambdaException

This exception is thrown when the Amazon Cognito service encounters an unexpected exception with the AWS Lambda service.

HTTP Status Code: 400

UserLambdaValidationException

This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user is not confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

SetRiskConfiguration

Configures actions on detected risks. To delete the risk configuration for `UserPoolId` or `ClientId`, pass null values for all four configuration types.

To enable Amazon Cognito advanced security features, update the user pool to include the `UserPoolAddOns` key`AdvancedSecurityMode`.

See UpdateUserPool (p. 288).

Request Syntax

```
{
   "AccountTakeoverRiskConfiguration": {
      "Actions": {
         "HighAction": {
            "EventAction": "string",
            "Notify": boolean
         },
         "LowAction": {
            "EventAction": "string",
            "Notify": boolean
         },
         "MediumAction": {
            "EventAction": "string",
            "Notify": boolean
         }
      },
      "NotifyConfiguration": {
         "BlockEmail": {
            "HtmlBody": "string",
            "Subject": "string",
            "TextBody": "string"
         },
         "From": "string",
         "MfaEmail": {
            "HtmlBody": "string",
            "Subject": "string",
            "TextBody": "string"
         },
         "NoActionEmail": {
            "HtmlBody": "string",
            "Subject": "string",
            "TextBody": "string"
         },
         "ReplyTo": "string",
         "SourceArn": "string"
      }
   },
   "ClientId": "string",
   "CompromisedCredentialsRiskConfiguration": {
      "Actions": {
         "EventAction": "string"
      },
      "EventFilter": [ "string" ]
   },
   "RiskExceptionConfiguration": {
      "BlockedIPRangeList": [ "string" ],
      "SkippedIPRangeList": [ "string" ]
   },
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AccountTakeoverRiskConfiguration (p. 243)

The account takeover risk configuration.

Type: AccountTakeoverRiskConfigurationType (p. 311) object

Required: No

ClientId (p. 243)

The app client ID. If `ClientId` is null, then the risk configuration is mapped to `userPoolId`. When the client ID is null, the same risk configuration is applied to all the clients in the userPool.

Otherwise, `ClientId` is mapped to the client. When the client ID is not null, the user pool configuration is overridden and the risk configuration for the client is used instead.

Type: String

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

Pattern: `[\w+]+`

Required: No

CompromisedCredentialsRiskConfiguration (p. 243)

The compromised credentials risk configuration.

Type: CompromisedCredentialsRiskConfigurationType (p. 323) object

Required: No

RiskExceptionConfiguration (p. 243)

The configuration to override the risk decision.

Type: RiskExceptionConfigurationType (p. 359) object

Required: No

UserPoolId (p. 243)

The user pool ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
```

```
   "RiskConfiguration": {
      "AccountTakeoverRiskConfiguration": {
         "Actions": {
            "HighAction": {
               "EventAction": "string",
               "Notify": boolean
            },
            "LowAction": {
               "EventAction": "string",
               "Notify": boolean
            },
            "MediumAction": {
               "EventAction": "string",
               "Notify": boolean
            }
         },
         "NotifyConfiguration": {
            "BlockEmail": {
               "HtmlBody": "string",
               "Subject": "string",
               "TextBody": "string"
            },
            "From": "string",
            "MfaEmail": {
               "HtmlBody": "string",
               "Subject": "string",
               "TextBody": "string"
            },
            "NoActionEmail": {
               "HtmlBody": "string",
               "Subject": "string",
               "TextBody": "string"
            },
            "ReplyTo": "string",
            "SourceArn": "string"
         }
      },
      "ClientId": "string",
      "CompromisedCredentialsRiskConfiguration": {
         "Actions": {
            "EventAction": "string"
         },
         "EventFilter": [ "string" ]
      },
      "LastModifiedDate": number,
      "RiskExceptionConfiguration": {
         "BlockedIPRangeList": [ "string" ],
         "SkippedIPRangeList": [ "string" ]
      },
      "UserPoolId": "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.

RiskConfiguration (p. 244)

The risk configuration.

Type: RiskConfigurationType (p. 357) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

CodeDeliveryFailureException

This exception is thrown when a verification code fails to deliver successfully.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidEmailRoleAccessPolicyException

This exception is thrown when Amazon Cognito is not allowed to use your email identity. HTTP status code: 400.

HTTP Status Code: 400

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserPoolAddOnNotEnabledException

This exception is thrown when user pool add-ons are not enabled.

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

SetUICustomization

Sets the UI customization information for a user pool's built-in app UI.

You can specify app UI customization settings for a single client (with a specific `clientId`) or for all clients (by setting the `clientId` to `ALL`). If you specify `ALL`, the default configuration will be used for every client that has no UI customization set previously. If you specify UI customization settings for a particular client, it will no longer fall back to the `ALL` configuration.

Note
To use this API, your user pool must have a domain associated with it. Otherwise, there is no place to host the app's pages, and the service will throw an error.

Request Syntax

```
{
   "ClientId": "string",
   "CSS": "string",
   "ImageFile": blob,
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

ClientId (p. 248)

The client ID for the client app.

Type: String

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

Pattern: `[\w+]+`

Required: No

CSS (p. 248)

The CSS values in the UI customization.

Type: String

Required: No

ImageFile (p. 248)

The uploaded logo image for the UI customization.

Type: Base64-encoded binary data object

Required: No

UserPoolId (p. 248)

The user pool ID for the user pool.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "UICustomization": {
      "ClientId": "string",
      "CreationDate": number,
      "CSS": "string",
      "CSSVersion": "string",
      "ImageUrl": "string",
      "LastModifiedDate": number,
      "UserPoolId": "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.

UICustomization (p. 249)

The UI customization information.

Type: UICustomizationType (p. 368) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

SetUserMFAPreference

Set the user's multi-factor authentication (MFA) method preference.

Request Syntax

```
{
   "AccessToken": "string",
   "SMSMfaSettings": {
      "Enabled": boolean,
      "PreferredMfa": boolean
   },
   "SoftwareTokenMfaSettings": {
      "Enabled": boolean,
      "PreferredMfa": boolean
   }
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AccessToken (p. 251)

The access token.

Type: String

Pattern: `[A-Za-z0-9-_=.]+`

Required: Yes

SMSMfaSettings (p. 251)

The SMS text message multi-factor authentication (MFA) settings.

Type: SMSMfaSettingsType (p. 364) object

Required: No

SoftwareTokenMfaSettings (p. 251)

The time-based one-time password software token MFA settings.

Type: SoftwareTokenMfaSettingsType (p. 366) object

Required: No

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user is not confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

SetUserPoolMfaConfig

Set the user pool MFA configuration.

Request Syntax

```
{
   "MfaConfiguration": "string",
   "SmsMfaConfiguration": {
      "SmsAuthenticationMessage": "string",
      "SmsConfiguration": {
         "ExternalId": "string",
         "SnsCallerArn": "string"
      }
   },
   "SoftwareTokenMfaConfiguration": {
      "Enabled": boolean
   },
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

MfaConfiguration (p. 253)

The MFA configuration.

Type: String

Valid Values: `OFF | ON | OPTIONAL`

Required: No

SmsMfaConfiguration (p. 253)

The SMS text message MFA configuration.

Type: SmsMfaConfigType (p. 363) object

Required: No

SoftwareTokenMfaConfiguration (p. 253)

The software token MFA configuration.

Type: SoftwareTokenMfaConfigType (p. 365) object

Required: No

UserPoolId (p. 253)

The user pool ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "MfaConfiguration": "string",
   "SmsMfaConfiguration": {
      "SmsAuthenticationMessage": "string",
      "SmsConfiguration": {
         "ExternalId": "string",
         "SnsCallerArn": "string"
      }
   },
   "SoftwareTokenMfaConfiguration": {
      "Enabled": boolean
   }
}
```

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.

MfaConfiguration (p. 254)

The MFA configuration.

Type: String

Valid Values: `OFF | ON | OPTIONAL`

SmsMfaConfiguration (p. 254)

The SMS text message MFA configuration.

Type: SmsMfaConfigType (p. 363) object

SoftwareTokenMfaConfiguration (p. 254)

The software token MFA configuration.

Type: SoftwareTokenMfaConfigType (p. 365) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidSmsRoleAccessPolicyException

This exception is returned when the role provided for SMS configuration does not have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is invalid for the role provided for SMS configuration. This can happen if you do not trust **cognito-idp.amazonaws.com** or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

SetUserSettings

Sets the user settings like multi-factor authentication (MFA). If MFA is to be removed for a particular attribute pass the attribute with code delivery as null. If null list is passed, all MFA options are removed.

Request Syntax

```
{
   "AccessToken": "string",
   "MFAOptions": [
      {
         "AttributeName": "string",
         "DeliveryMedium": "string"
      }
   ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AccessToken (p. 256)

The access token for the set user settings request.

Type: String

Pattern: `[A-Za-z0-9-_=.]+`

Required: Yes

MFAOptions (p. 256)

Specifies the options for MFA (e.g., email or phone number).

Type: Array of MFAOptionType (p. 344) objects

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user is not confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

SignUp

Registers the user in the specified user pool and creates a user name, password, and user attributes.

Request Syntax

```
{
   "AnalyticsMetadata": {
      "AnalyticsEndpointId": "string"
   },
   "ClientId": "string",
   "Password": "string",
   "SecretHash": "string",
   "UserAttributes": [
      {
         "Name": "string",
         "Value": "string"
      }
   ],
   "UserContextData": {
      "EncodedData": "string"
   },
   "Username": "string",
   "ValidationData": [
      {
         "Name": "string",
         "Value": "string"
      }
   ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AnalyticsMetadata (p. 258)

The Amazon Pinpoint analytics metadata for collecting metrics for `SignUp` calls.

Type: AnalyticsMetadataType (p. 314) object

Required: No

ClientId (p. 258)

The ID of the client associated with the user pool.

Type: String

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

Pattern: `[\w+]+`

Required: Yes

Password (p. 258)

The password of the user you wish to register.

Type: String

Length Constraints: Minimum length of 6. Maximum length of 256.

Pattern: `[\S]+`

Required: Yes

SecretHash (p. 258)

A keyed-hash message authentication code (HMAC) calculated using the secret key of a user pool client and username plus the client ID in the message.

Type: String

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

Pattern: `[\w+=/]+`

Required: No

UserAttributes (p. 258)

An array of name-value pairs representing user attributes.

For custom attributes, you must prepend the `custom:` prefix to the attribute name.

Type: Array of AttributeType (p. 315) objects

Required: No

UserContextData (p. 258)

Contextual data such as the user's device fingerprint, IP address, or location used for evaluating the risk of an unexpected event by Amazon Cognito advanced security.

Type: UserContextDataType (p. 370) object

Required: No

Username (p. 258)

The user name of the user you wish to register.

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

ValidationData (p. 258)

The validation data in the request to register a user.

Type: Array of AttributeType (p. 315) objects

Required: No

Response Syntax

```
{
   "CodeDeliveryDetails": {
      "AttributeName": "string",
```

```
      "DeliveryMedium": "string",
      "Destination": "string"
   },
   "UserConfirmed": boolean,
   "UserSub": "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.

CodeDeliveryDetails (p. 259)

The code delivery details returned by the server response to the user registration request.

Type: CodeDeliveryDetailsType (p. 321) object

UserConfirmed (p. 259)

A response from the server indicating that a user registration has been confirmed.

Type: Boolean

UserSub (p. 259)

The UUID of the authenticated user. This is not the same as `username`.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

CodeDeliveryFailureException

This exception is thrown when a verification code fails to deliver successfully.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidEmailRoleAccessPolicyException

This exception is thrown when Amazon Cognito is not allowed to use your email identity. HTTP status code: 400.

HTTP Status Code: 400

InvalidLambdaResponseException

This exception is thrown when the Amazon Cognito service encounters an invalid AWS Lambda response.

HTTP Status Code: 400

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidPasswordException

This exception is thrown when the Amazon Cognito service encounters an invalid password.

HTTP Status Code: 400

InvalidSmsRoleAccessPolicyException

This exception is returned when the role provided for SMS configuration does not have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is invalid for the role provided for SMS configuration. This can happen if you do not trust **cognito-idp.amazonaws.com** or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UnexpectedLambdaException

This exception is thrown when the Amazon Cognito service encounters an unexpected exception with the AWS Lambda service.

HTTP Status Code: 400

UserLambdaValidationException

This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.

HTTP Status Code: 400

UsernameExistsException

This exception is thrown when Amazon Cognito encounters a user name that already exists in the user pool.

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

StartUserImportJob

Starts the user import.

Request Syntax

```
{
   "JobId": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

JobId (p. 263)

The job ID for the user import job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `import-[0-9a-zA-Z-]+`

Required: Yes

UserPoolId (p. 263)

The user pool ID for the user pool that the users are being imported into.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "UserImportJob": {
      "CloudWatchLogsRoleArn": "string",
      "CompletionDate": number,
      "CompletionMessage": "string",
      "CreationDate": number,
      "FailedUsers": number,
      "ImportedUsers": number,
      "JobId": "string",
      "JobName": "string",
      "PreSignedUrl": "string",
      "SkippedUsers": number,
      "StartDate": number,
```

```
      "Status": "string",
      "UserPoolId": "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.

UserImportJob (p. 263)

The job object that represents the user import job.

Type: UserImportJobType (p. 371) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

PreconditionNotMetException

This exception is thrown when a precondition is not met.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

StopUserImportJob

Stops the user import job.

Request Syntax

```
{
   "JobId": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

JobId (p. 266)

The job ID for the user import job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `import-[0-9a-zA-Z-]+`

Required: Yes

UserPoolId (p. 266)

The user pool ID for the user pool that the users are being imported into.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "UserImportJob": {
      "CloudWatchLogsRoleArn": "string",
      "CompletionDate": number,
      "CompletionMessage": "string",
      "CreationDate": number,
      "FailedUsers": number,
      "ImportedUsers": number,
      "JobId": "string",
      "JobName": "string",
      "PreSignedUrl": "string",
      "SkippedUsers": number,
      "StartDate": number,
```

```
      "Status": "string",
      "UserPoolId": "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.

UserImportJob (p. 266)

The job object that represents the user import job.

Type: UserImportJobType (p. 371) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

PreconditionNotMetException

This exception is thrown when a precondition is not met.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

UpdateAuthEventFeedback

Provides the feedback for an authentication event whether it was from a valid user or not. This feedback is used for improving the risk evaluation decision for the user pool as part of Amazon Cognito advanced security.

Request Syntax

```
{
   "EventId": "string",
   "FeedbackToken": "string",
   "FeedbackValue": "string",
   "Username": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

EventId (p. 269)

The event ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 50.

Pattern: `[\w+-]+`

Required: Yes

FeedbackToken (p. 269)

The feedback token.

Type: String

Pattern: `[A-Za-z0-9-_=.]+`

Required: Yes

FeedbackValue (p. 269)

The authentication event feedback value.

Type: String

Valid Values: `Valid | Invalid`

Required: Yes

Username (p. 269)

The user pool username.

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

UserPoolId (p. 269)

The user pool ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

HTTP Status Code: 400

UserPoolAddOnNotEnabledException

This exception is thrown when user pool add-ons are not enabled.

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

UpdateDeviceStatus

Updates the device status.

Request Syntax

```
{
   "AccessToken": "string",
   "DeviceKey": "string",
   "DeviceRememberedStatus": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AccessToken (p. 272)

The access token.

Type: String

Pattern: `[A-Za-z0-9-_=.]+`

Required: Yes

DeviceKey (p. 272)

The device key.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-f-]+`

Required: Yes

DeviceRememberedStatus (p. 272)

The status of whether a device is remembered.

Type: String

Valid Values: `remembered | not_remembered`

Required: No

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidUserPoolConfigurationException

This exception is thrown when the user pool configuration is invalid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user is not confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

UpdateGroup

Updates the specified group with the specified attributes.

Requires developer credentials.

Request Syntax

```
{
   "Description": "string",
   "GroupName": "string",
   "Precedence": number,
   "RoleArn": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

Description (p. 275)

A string containing the new description of the group.

Type: String

Length Constraints: Maximum length of 2048.

Required: No

GroupName (p. 275)

The name of the group.

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

Precedence (p. 275)

The new precedence value for the group. For more information about this parameter, see CreateGroup (p. 101).

Type: Integer

Valid Range: Minimum value of 0.

Required: No

RoleArn (p. 275)

The new role ARN for the group. This is used for setting the `cognito:roles` and `cognito:preferred_role` claims in the token.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:([\w+=/,.@-]*)?:[0-9]+:[\w+=/,.@-]+(:[\w+=/,.@-]+)?(:[\w+=/,.@-]+)?`

Required: No

UserPoolId (p. 275)

The user pool ID for the user pool.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "Group": {
      "CreationDate": number,
      "Description": "string",
      "GroupName": "string",
      "LastModifiedDate": number,
      "Precedence": number,
      "RoleArn": "string",
      "UserPoolId": "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.

Group (p. 276)

The group object for the group.

Type: GroupType (p. 335) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

UpdateIdentityProvider

Updates identity provider information for a user pool.

Request Syntax

```
{
   "AttributeMapping": {
      "string" : "string"
   },
   "IdpIdentifiers": [ "string" ],
   "ProviderDetails": {
      "string" : "string"
   },
   "ProviderName": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AttributeMapping (p. 278)

The identity provider attribute mapping to be changed.

Type: String to string map

Key Length Constraints: Minimum length of 1. Maximum length of 32.

Required: No

IdpIdentifiers (p. 278)

A list of identity provider identifiers.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 50 items.

Length Constraints: Minimum length of 1. Maximum length of 40.

Pattern: `[\w\s+=.@-]+`

Required: No

ProviderDetails (p. 278)

The identity provider details to be updated, such as `MetadataURL` and `MetadataFile`.

Type: String to string map

Required: No

ProviderName (p. 278)

The identity provider name.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 32.

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

Required: Yes

UserPoolId (p. 278)

The user pool ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "IdentityProvider": {
      "AttributeMapping": {
         "string" : "string"
      },
      "CreationDate": number,
      "IdpIdentifiers": [ "string" ],
      "LastModifiedDate": number,
      "ProviderDetails": {
         "string" : "string"
      },
      "ProviderName": "string",
      "ProviderType": "string",
      "UserPoolId": "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.

IdentityProvider (p. 279)

The identity provider object.

Type: IdentityProviderType (p. 338) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UnsupportedIdentityProviderException

This exception is thrown when the specified identifier is not supported.

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

UpdateResourceServer

Updates the name and scopes of resource server. All other fields are read-only.

Request Syntax

```
{
   "Identifier": "string",
   "Name": "string",
   "Scopes": [
      {
         "ScopeDescription": "string",
         "ScopeName": "string"
      }
   ],
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

Identifier (p. 281)

The identifier for the resource server.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `[\x21\x23-\x5B\x5D-\x7E]+`

Required: Yes

Name (p. 281)

The name of the resource server.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `[\w\s+=,.@-]+`

Required: Yes

Scopes (p. 281)

The scope values to be set for the resource server.

Type: Array of ResourceServerScopeType (p. 354) objects

Array Members: Maximum number of 25 items.

Required: No

UserPoolId (p. 281)

The user pool ID for the user pool.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
   "ResourceServer": {
      "Identifier": "string",
      "Name": "string",
      "Scopes": [
         {
            "ScopeDescription": "string",
            "ScopeName": "string"
         }
      ],
      "UserPoolId": "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.

ResourceServer (p. 282)

The resource server.

Type: ResourceServerType (p. 355) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

UpdateUserAttributes

Allows a user to update a specific attribute (one at a time).

Request Syntax

```
{
   "AccessToken": "string",
   "UserAttributes": [
      {
         "Name": "string",
         "Value": "string"
      }
   ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AccessToken (p. 284)

The access token for the request to update user attributes.

Type: String

Pattern: `[A-Za-z0-9-_=.]+`

Required: Yes

UserAttributes (p. 284)

An array of name-value pairs representing user attributes.

For custom attributes, you must prepend the `custom:` prefix to the attribute name.

Type: Array of AttributeType (p. 315) objects

Required: Yes

Response Syntax

```
{
   "CodeDeliveryDetailsList": [
      {
         "AttributeName": "string",
         "DeliveryMedium": "string",
         "Destination": "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.

CodeDeliveryDetailsList (p. 284)

The code delivery details list from the server for the request to update user attributes.

Type: Array of CodeDeliveryDetailsType (p. 321) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

AliasExistsException

This exception is thrown when a user tries to confirm the account with an email or phone number that has already been supplied as an alias from a different account. This exception tells user that an account with this email or phone already exists.

HTTP Status Code: 400

CodeDeliveryFailureException

This exception is thrown when a verification code fails to deliver successfully.

HTTP Status Code: 400

CodeMismatchException

This exception is thrown if the provided code does not match what the server was expecting.

HTTP Status Code: 400

ExpiredCodeException

This exception is thrown if a code has expired.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidEmailRoleAccessPolicyException

This exception is thrown when Amazon Cognito is not allowed to use your email identity. HTTP status code: 400.

HTTP Status Code: 400

InvalidLambdaResponseException

This exception is thrown when the Amazon Cognito service encounters an invalid AWS Lambda response.

HTTP Status Code: 400

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidSmsRoleAccessPolicyException

This exception is returned when the role provided for SMS configuration does not have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is invalid for the role provided for SMS configuration. This can happen if you do not trust **cognito-idp.amazonaws.com** or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UnexpectedLambdaException

This exception is thrown when the Amazon Cognito service encounters an unexpected exception with the AWS Lambda service.

HTTP Status Code: 400

UserLambdaValidationException

This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user is not confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

UpdateUserPool

Updates the specified user pool with the specified attributes. If you don't provide a value for an attribute, it will be set to the default value. You can get a list of the current user pool settings with DescribeUserPool (p. 158).

Request Syntax

```
{
   "AdminCreateUserConfig": {
      "AllowAdminCreateUserOnly": boolean,
      "InviteMessageTemplate": {
         "EmailMessage": "string",
         "EmailSubject": "string",
         "SMSMessage": "string"
      },
      "UnusedAccountValidityDays": number
   },
   "AutoVerifiedAttributes": [ "string" ],
   "DeviceConfiguration": {
      "ChallengeRequiredOnNewDevice": boolean,
      "DeviceOnlyRememberedOnUserPrompt": boolean
   },
   "EmailConfiguration": {
      "ReplyToEmailAddress": "string",
      "SourceArn": "string"
   },
   "EmailVerificationMessage": "string",
   "EmailVerificationSubject": "string",
   "LambdaConfig": {
      "CreateAuthChallenge": "string",
      "CustomMessage": "string",
      "DefineAuthChallenge": "string",
      "PostAuthentication": "string",
      "PostConfirmation": "string",
      "PreAuthentication": "string",
      "PreSignUp": "string",
      "PreTokenGeneration": "string",
      "UserMigration": "string",
      "VerifyAuthChallengeResponse": "string"
   },
   "MfaConfiguration": "string",
   "Policies": {
      "PasswordPolicy": {
         "MinimumLength": number,
         "RequireLowercase": boolean,
         "RequireNumbers": boolean,
         "RequireSymbols": boolean,
         "RequireUppercase": boolean
      }
   },
   "SmsAuthenticationMessage": "string",
   "SmsConfiguration": {
      "ExternalId": "string",
      "SnsCallerArn": "string"
   },
   "SmsVerificationMessage": "string",
   "UserPoolAddOns": {
      "AdvancedSecurityMode": "string"
   },
   "UserPoolId": "string",
   "UserPoolTags": {
```

```
      "string" : "string"
   },
   "VerificationMessageTemplate": {
      "DefaultEmailOption": "string",
      "EmailMessage": "string",
      "EmailMessageByLink": "string",
      "EmailSubject": "string",
      "EmailSubjectByLink": "string",
      "SmsMessage": "string"
   }
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AdminCreateUserConfig (p. 288)

The configuration for `AdminCreateUser` requests.

Type: AdminCreateUserConfigType (p. 312) object

Required: No

AutoVerifiedAttributes (p. 288)

The attributes that are automatically verified when the Amazon Cognito service makes a request to update user pools.

Type: Array of strings

Valid Values: `phone_number | email`

Required: No

DeviceConfiguration (p. 288)

Device configuration.

Type: DeviceConfigurationType (p. 326) object

Required: No

EmailConfiguration (p. 288)

Email configuration.

Type: EmailConfigurationType (p. 331) object

Required: No

EmailVerificationMessage (p. 288)

The contents of the email verification message.

Type: String

Length Constraints: Minimum length of 6. Maximum length of 20000.

Pattern: `[\p{L}\p{M}\p{S}\p{N}\p{P}\s*]*\{####\}[\p{L}\p{M}\p{S}\p{N}\p{P}\s*]*`

Required: No

EmailVerificationSubject (p. 288)

The subject of the email verification message.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 140.

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

Required: No

LambdaConfig (p. 288)

The AWS Lambda configuration information from the request to update the user pool.

Type: LambdaConfigType (p. 340) object

Required: No

MfaConfiguration (p. 288)

Can be one of the following values:

- `OFF` - MFA tokens are not required and cannot be specified during user registration.
- `ON` - MFA tokens are required for all user registrations. You can only specify required when you are initially creating a user pool.
- `OPTIONAL` - Users have the option when registering to create an MFA token.

Type: String

Valid Values: `OFF | ON | OPTIONAL`

Required: No

Policies (p. 288)

A container with the policies you wish to update in a user pool.

Type: UserPoolPolicyType (p. 382) object

Required: No

SmsAuthenticationMessage (p. 288)

The contents of the SMS authentication message.

Type: String

Length Constraints: Minimum length of 6. Maximum length of 140.

Pattern: `.*\{####\}.*`

Required: No

SmsConfiguration (p. 288)

SMS configuration.

Type: SmsConfigurationType (p. 362) object

Required: No

SmsVerificationMessage (p. 288)

A container with information about the SMS verification message.

Type: String

Length Constraints: Minimum length of 6. Maximum length of 140.

Pattern: `.*\{####\}.*`

Required: No

UserPoolAddOns (p. 288)

Used to enable advanced security risk detection. Set the key `AdvancedSecurityMode` to the value "AUDIT".

Type: UserPoolAddOnsType (p. 374) object

Required: No

UserPoolId (p. 288)

The user pool ID for the user pool you want to update.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

UserPoolTags (p. 288)

The cost allocation tags for the user pool. For more information, see Adding Cost Allocation Tags to Your User Pool

Type: String to string map

Required: No

VerificationMessageTemplate (p. 288)

The template for verification messages.

Type: VerificationMessageTemplateType (p. 390) object

Required: No

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

ConcurrentModificationException

This exception is thrown if two or more modifications are happening concurrently.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidEmailRoleAccessPolicyException

This exception is thrown when Amazon Cognito is not allowed to use your email identity. HTTP status code: 400.

HTTP Status Code: 400

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidSmsRoleAccessPolicyException

This exception is returned when the role provided for SMS configuration does not have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is invalid for the role provided for SMS configuration. This can happen if you do not trust **cognito-idp.amazonaws.com** or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserImportInProgressException

This exception is thrown when you are trying to modify a user pool while a user import job is in progress for that pool.

HTTP Status Code: 400

UserPoolTaggingException

This exception is thrown when a user pool tag cannot be set or updated.

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

UpdateUserPoolClient

Updates the specified user pool app client with the specified attributes. If you don't provide a value for an attribute, it will be set to the default value. You can get a list of the current user pool app client settings with DescribeUserPoolClient (p. 162).

Request Syntax

```
{
   "AllowedOAuthFlows": [ "string" ],
   "AllowedOAuthFlowsUserPoolClient": boolean,
   "AllowedOAuthScopes": [ "string" ],
   "AnalyticsConfiguration": {
      "ApplicationId": "string",
      "ExternalId": "string",
      "RoleArn": "string",
      "UserDataShared": boolean
   },
   "CallbackURLs": [ "string" ],
   "ClientId": "string",
   "ClientName": "string",
   "DefaultRedirectURI": "string",
   "ExplicitAuthFlows": [ "string" ],
   "LogoutURLs": [ "string" ],
   "ReadAttributes": [ "string" ],
   "RefreshTokenValidity": number,
   "SupportedIdentityProviders": [ "string" ],
   "UserPoolId": "string",
   "WriteAttributes": [ "string" ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AllowedOAuthFlows (p. 294)

Set to `code` to initiate a code grant flow, which provides an authorization code as the response. This code can be exchanged for access tokens with the token endpoint.

Set to `token` to specify that the client should get the access token (and, optionally, ID token, based on scopes) directly.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 3 items.

Valid Values: `code | implicit | client_credentials`

Required: No

AllowedOAuthFlowsUserPoolClient (p. 294)

Set to TRUE if the client is allowed to follow the OAuth protocol when interacting with Cognito user pools.

Type: Boolean

Required: No

AllowedOAuthScopes (p. 294)

A list of allowed `OAuth` scopes. Currently supported values are `"phone"`, `"email"`, `"openid"`, and `"Cognito"`.

Type: Array of strings

Array Members: Maximum number of 25 items.

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `[\x21\x23-\x5B\x5D-\x7E]+`

Required: No

AnalyticsConfiguration (p. 294)

The Amazon Pinpoint analytics configuration for collecting metrics for this user pool.

Type: AnalyticsConfigurationType (p. 313) object

Required: No

CallbackURLs (p. 294)

A list of allowed redirect (callback) URLs for the identity providers.

A redirect URI must:

- Be an absolute URI.
- Be registered with the authorization server.
- Not include a fragment component.

See OAuth 2.0 - Redirection Endpoint.

Amazon Cognito requires HTTPS over HTTP except for http://localhost for testing purposes only.

App callback URLs such as myapp://example are also supported.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 100 items.

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

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

Required: No

ClientId (p. 294)

The ID of the client associated with the user pool.

Type: String

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

Pattern: `[\w+]+`

Required: Yes

ClientName (p. 294)

The client name from the update user pool client request.

Type: String

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

Pattern: `[\w\s+=,.@-]+`

Required: No

DefaultRedirectURI (p. 294)

The default redirect URI. Must be in the `CallbackURLs` list.

A redirect URI must:

- Be an absolute URI.
- Be registered with the authorization server.
- Not include a fragment component.

See OAuth 2.0 - Redirection Endpoint.

Amazon Cognito requires HTTPS over HTTP except for http://localhost for testing purposes only.

App callback URLs such as myapp://example are also supported.

Type: String

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

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

Required: No

ExplicitAuthFlows (p. 294)

Explicit authentication flows.

Type: Array of strings

Valid Values: `ADMIN_NO_SRP_AUTH | CUSTOM_AUTH_FLOW_ONLY | USER_PASSWORD_AUTH`

Required: No

LogoutURLs (p. 294)

A list of allowed logout URLs for the identity providers.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 100 items.

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

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

Required: No

ReadAttributes (p. 294)

The read-only attributes of the user pool.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: No

RefreshTokenValidity (p. 294)

The time limit, in days, after which the refresh token is no longer valid and cannot be used.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 3650.

Required: No

SupportedIdentityProviders (p. 294)

A list of provider names for the identity providers that are supported on this client.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 32.

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

Required: No

UserPoolId (p. 294)

The user pool ID for the user pool where you want to update the user pool client.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

WriteAttributes (p. 294)

The writeable attributes of the user pool.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: No

Response Syntax

```
{
   "UserPoolClient": {
      "AllowedOAuthFlows": [ "string" ],
      "AllowedOAuthFlowsUserPoolClient": boolean,
      "AllowedOAuthScopes": [ "string" ],
      "AnalyticsConfiguration": {
         "ApplicationId": "string",
         "ExternalId": "string",
         "RoleArn": "string",
         "UserDataShared": boolean
      },
      "CallbackURLs": [ "string" ],
      "ClientId": "string",
      "ClientName": "string",
      "ClientSecret": "string",
      "CreationDate": number,
      "DefaultRedirectURI": "string",
```

```
      "ExplicitAuthFlows": [ "string" ],
      "LastModifiedDate": number,
      "LogoutURLs": [ "string" ],
      "ReadAttributes": [ "string" ],
      "RefreshTokenValidity": number,
      "SupportedIdentityProviders": [ "string" ],
      "UserPoolId": "string",
      "WriteAttributes": [ "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.

UserPoolClient (p. 297)

The user pool client value from the response from the server when an update user pool client request is made.

Type: UserPoolClientType (p. 376) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

ConcurrentModificationException

This exception is thrown if two or more modifications are happening concurrently.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidOAuthFlowException

This exception is thrown when the specified OAuth flow is invalid.

HTTP Status Code: 400

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

ScopeDoesNotExistException

This exception is thrown when the specified scope does not exist.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

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

VerifySoftwareToken

Use this API to register a user's entered TOTP code and mark the user's software token MFA status as "verified" if successful. The request takes an access token or a session string, but not both.

Request Syntax

```
{
   "AccessToken": "string",
   "FriendlyDeviceName": "string",
   "Session": "string",
   "UserCode": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AccessToken (p. 300)

The access token.

Type: String

Pattern: `[A-Za-z0-9-_=.]+`

Required: No

FriendlyDeviceName (p. 300)

The friendly device name.

Type: String

Required: No

Session (p. 300)

The session which should be passed both ways in challenge-response calls to the service.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Required: No

UserCode (p. 300)

The one time password computed using the secret code returned by AssociateSoftwareToken (p. 84)

Type: String

Length Constraints: Fixed length of 6.

Pattern: `[0-9]+`

Required: Yes

Response Syntax

```
{
   "Session": "string",
   "Status": "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.

Session (p. 301)

The session which should be passed both ways in challenge-response calls to the service.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Status (p. 301)

The status of the verify software token.

Type: String

Valid Values: `SUCCESS | ERROR`

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

CodeMismatchException

This exception is thrown if the provided code does not match what the server was expecting.

HTTP Status Code: 400

EnableSoftwareTokenMFAException

This exception is thrown when there is a code mismatch and the service fails to configure the software token TOTP multi-factor authentication (MFA).

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidUserPoolConfigurationException

This exception is thrown when the user pool configuration is invalid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

SoftwareTokenMFANotFoundException

This exception is thrown when the software token TOTP multi-factor authentication (MFA) is not enabled for the user pool.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user is not confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

VerifyUserAttribute

Verifies the specified user attributes in the user pool.

Request Syntax

```
{
   "AccessToken": "string",
   "AttributeName": "string",
   "Code": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 392).

The request accepts the following data in JSON format.

AccessToken (p. 304)

Represents the access token of the request to verify user attributes.

Type: String

Pattern: `[A-Za-z0-9-_=.]+`

Required: Yes

AttributeName (p. 304)

The attribute name in the request to verify user attributes.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 32.

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

Required: Yes

Code (p. 304)

The verification code in the request to verify user attributes.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `[\S]+`

Required: Yes

Response Elements

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 394).

CodeMismatchException

This exception is thrown if the provided code does not match what the server was expecting.

HTTP Status Code: 400

ExpiredCodeException

This exception is thrown if a code has expired.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

LimitExceededException

This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user is not authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service cannot find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user is not confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user is not found.

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

Data Types

The Amazon Cognito Identity Provider 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:

AccountTakeoverActionsType

Account takeover actions type.

Contents

HighAction

Action to take for a high risk.

Type: AccountTakeoverActionType (p. 310) object

Required: No

LowAction

Action to take for a low risk.

Type: AccountTakeoverActionType (p. 310) object

Required: No

MediumAction

Action to take for a medium risk.

Type: AccountTakeoverActionType (p. 310) object

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

AccountTakeoverActionType

Account takeover action type.

Contents

EventAction

The event action.

- `BLOCK` Choosing this action will block the request.
- `MFA_IF_CONFIGURED` Throw MFA challenge if user has configured it, else allow the request.
- `MFA_REQUIRED` Throw MFA challenge if user has configured it, else block the request.
- `NO_ACTION` Allow the user sign-in.

Type: String

Valid Values: `BLOCK | MFA_IF_CONFIGURED | MFA_REQUIRED | NO_ACTION`

Required: Yes

Notify

Flag specifying whether to send a notification.

Type: Boolean

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

AccountTakeoverRiskConfigurationType

Configuration for mitigation actions and notification for different levels of risk detected for a potential account takeover.

Contents

Actions

Account takeover risk configuration actions

Type: AccountTakeoverActionsType (p. 309) object

Required: Yes

NotifyConfiguration

The notify configuration used to construct email notifications.

Type: NotifyConfigurationType (p. 346) object

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

AdminCreateUserConfigType

The configuration for creating a new user profile.

Contents

AllowAdminCreateUserOnly

Set to `True` if only the administrator is allowed to create user profiles. Set to `False` if users can sign themselves up via an app.

Type: Boolean

Required: No

InviteMessageTemplate

The message template to be used for the welcome message to new users.

See also Customizing User Invitation Messages.

Type: MessageTemplateType (p. 343) object

Required: No

UnusedAccountValidityDays

The user account expiration limit, in days, after which the account is no longer usable. To reset the account after that time limit, you must call `AdminCreateUser` again, specifying `"RESEND"` for the `MessageAction` parameter. The default value for this parameter is 7.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 365.

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

AnalyticsConfigurationType

The Amazon Pinpoint analytics configuration for collecting metrics for a user pool.

Contents

ApplicationId

The application ID for an Amazon Pinpoint application.

Type: String

Pattern: `^[0-9a-fA-F]+$`

Required: Yes

ExternalId

The external ID.

Type: String

Required: Yes

RoleArn

The ARN of an IAM role that authorizes Amazon Cognito to publish events to Amazon Pinpoint analytics.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:([\w+=/,.@-]*)?:[0-9]+:[\w+=/,.@-]+(:[\w+=/,.@-]+)?(:[\w+=/,.@-]+)?`

Required: Yes

UserDataShared

If `UserDataShared` is `true`, Amazon Cognito will include user data in the events it publishes to Amazon Pinpoint analytics.

Type: Boolean

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

AnalyticsMetadataType

An Amazon Pinpoint analytics endpoint.

An endpoint uniquely identifies a mobile device, email address, or phone number that can receive messages from Amazon Pinpoint analytics.

Contents

AnalyticsEndpointId

The endpoint ID.

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

AttributeType

Specifies whether the attribute is standard or custom.

Contents

Name

The name of the attribute.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 32.

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

Required: Yes

Value

The value of the attribute.

Type: String

Length Constraints: Maximum length of 2048.

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

AuthenticationResultType

The authentication result.

Contents

AccessToken

The access token.

Type: String

Pattern: `[A-Za-z0-9-_=.]+`

Required: No

ExpiresIn

The expiration period of the authentication result in seconds.

Type: Integer

Required: No

IdToken

The ID token.

Type: String

Pattern: `[A-Za-z0-9-_=.]+`

Required: No

NewDeviceMetadata

The new device metadata from an authentication result.

Type: NewDeviceMetadataType (p. 345) object

Required: No

RefreshToken

The refresh token.

Type: String

Pattern: `[A-Za-z0-9-_=.]+`

Required: No

TokenType

The token type.

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

AuthEventType

The authentication event type.

Contents

ChallengeResponses

The challenge responses.

Type: Array of ChallengeResponseType (p. 320) objects

Required: No

CreationDate

The creation date

Type: Timestamp

Required: No

EventContextData

The user context data captured at the time of an event request. It provides additional information about the client from which event the request is received.

Type: EventContextDataType (p. 332) object

Required: No

EventFeedback

A flag specifying the user feedback captured at the time of an event request is good or bad.

Type: EventFeedbackType (p. 333) object

Required: No

EventId

The event ID.

Type: String

Required: No

EventResponse

The event response.

Type: String

Valid Values: `Success | Failure`

Required: No

EventRisk

The event risk.

Type: EventRiskType (p. 334) object

Required: No

EventType

The event type.

Type: String

Valid Values: `SignIn | SignUp | ForgotPassword`

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

ChallengeResponseType

The challenge response type.

Contents

ChallengeName

The challenge name

Type: String

Valid Values: `Password | Mfa`

Required: No

ChallengeResponse

The challenge response.

Type: String

Valid Values: `Success | Failure`

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

CodeDeliveryDetailsType

The code delivery details being returned from the server.

Contents

AttributeName

The attribute name.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 32.

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

Required: No

DeliveryMedium

The delivery medium (email message or phone number).

Type: String

Valid Values: `SMS | EMAIL`

Required: No

Destination

The destination for the code delivery details.

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

CompromisedCredentialsActionsType

The compromised credentials actions type

Contents

EventAction

The event action.

Type: String

Valid Values: `BLOCK | NO_ACTION`

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

CompromisedCredentialsRiskConfigurationType

The compromised credentials risk configuration type.

Contents

Actions

The compromised credentials risk configuration actions.

Type: CompromisedCredentialsActionsType (p. 322) object

Required: Yes

EventFilter

Perform the action for these events. The default is to perform all events if no event filter is specified.

Type: Array of strings

Valid Values: `SIGN_IN | PASSWORD_CHANGE | SIGN_UP`

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

ContextDataType

Contextual user data type used for evaluating the risk of an unexpected event by Amazon Cognito advanced security.

Contents

EncodedData

Encoded data containing device fingerprinting details, collected using the Amazon Cognito context data collection library.

Type: String

Required: No

HttpHeaders

HttpHeaders received on your server in same order.

Type: Array of HttpHeader (p. 337) objects

Required: Yes

IpAddress

Source IP address of your user.

Type: String

Required: Yes

ServerName

Your server endpoint where this API is invoked.

Type: String

Required: Yes

ServerPath

Your server path where this API is invoked.

Type: String

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

CustomDomainConfigType

The configuration for a custom domain that hosts the sign-up and sign-in webpages for your application.

Contents

CertificateArn

The Amazon Resource Name (ARN) of an AWS Certificate Manager SSL certificate. You use this certificate for the subdomain of your custom domain.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:([\w+=/,.@-]*)?:[0-9]+:[\w+=/,.@-]+(:[\w+=/,.@-]+)?(:[\w+=/,.@-]+)?`

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

DeviceConfigurationType

The configuration for the user pool's device tracking.

Contents

ChallengeRequiredOnNewDevice

Indicates whether a challenge is required on a new device. Only applicable to a new device.

Type: Boolean

Required: No

DeviceOnlyRememberedOnUserPrompt

If true, a device is only remembered on user prompt.

Type: Boolean

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

DeviceSecretVerifierConfigType

The device verifier against which it will be authenticated.

Contents

PasswordVerifier

The password verifier.

Type: String

Required: No

Salt

The salt.

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

DeviceType

The device type.

Contents

DeviceAttributes

The device attributes.

Type: Array of AttributeType (p. 315) objects

Required: No

DeviceCreateDate

The creation date of the device.

Type: Timestamp

Required: No

DeviceKey

The device key.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-f-]+`

Required: No

DeviceLastAuthenticatedDate

The date in which the device was last authenticated.

Type: Timestamp

Required: No

DeviceLastModifiedDate

The last modified date of the device.

Type: Timestamp

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

DomainDescriptionType

A container for information about a domain.

Contents

AWSAccountId

The AWS account ID for the user pool owner.

Type: String

Required: No

CloudFrontDistribution

The ARN of the CloudFront distribution.

Type: String

Required: No

CustomDomainConfig

The configuration for a custom domain that hosts the sign-up and sign-in webpages for your application.

Type: CustomDomainConfigType (p. 325) object

Required: No

Domain

The domain string.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 63.

Pattern: `^[a-z0-9](?:[a-z0-9\-]{0,61}[a-z0-9])?$`

Required: No

S3Bucket

The S3 bucket where the static files for this domain are stored.

Type: String

Length Constraints: Minimum length of 3. Maximum length of 1024.

Pattern: `^[0-9A-Za-z\.\-_]*(?<!\.)$`

Required: No

Status

The domain status.

Type: String

Valid Values: `CREATING | DELETING | UPDATING | ACTIVE | FAILED`

Required: No

UserPoolId

The user pool ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: No

Version

The app version.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 20.

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

EmailConfigurationType

The email configuration type.

Contents

ReplyToEmailAddress

The destination to which the receiver of the email should reply to.

Type: String

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

Required: No

SourceArn

The Amazon Resource Name (ARN) of the email source.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:([\w+=/,.@-]*)?:[0-9]+:[\w+=/,.@-]+(:[\w+=/,.@-]+)?(:[\w+=/,.@-]+)?`

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

EventContextDataType

Specifies the user context data captured at the time of an event request.

Contents

City

The user's city.

Type: String

Required: No

Country

The user's country.

Type: String

Required: No

DeviceName

The user's device name.

Type: String

Required: No

IpAddress

The user's IP address.

Type: String

Required: No

Timezone

The user's time zone.

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

EventFeedbackType

Specifies the event feedback type.

Contents

FeedbackDate

The event feedback date.

Type: Timestamp

Required: No

FeedbackValue

The event feedback value.

Type: String

Valid Values: `Valid | Invalid`

Required: Yes

Provider

The provider.

Type: String

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

EventRiskType

The event risk type.

Contents

RiskDecision

The risk decision.

Type: String

Valid Values: `NoRisk | AccountTakeover | Block`

Required: No

RiskLevel

The risk level.

Type: String

Valid Values: `Low | Medium | High`

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

GroupType

The group type.

Contents

CreationDate

The date the group was created.

Type: Timestamp

Required: No

Description

A string containing the description of the group.

Type: String

Length Constraints: Maximum length of 2048.

Required: No

GroupName

The name of the group.

Type: String

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

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

Required: No

LastModifiedDate

The date the group was last modified.

Type: Timestamp

Required: No

Precedence

A nonnegative integer value that specifies the precedence of this group relative to the other groups that a user can belong to in the user pool. If a user belongs to two or more groups, it is the group with the highest precedence whose role ARN will be used in the `cognito:roles` and `cognito:preferred_role` claims in the user's tokens. Groups with higher `Precedence` values take precedence over groups with lower `Precedence` values or with null `Precedence` values.

Two groups can have the same `Precedence` value. If this happens, neither group takes precedence over the other. If two groups with the same `Precedence` have the same role ARN, that role is used in the `cognito:preferred_role` claim in tokens for users in each group. If the two groups have different role ARNs, the `cognito:preferred_role` claim is not set in users' tokens.

The default `Precedence` value is null.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

RoleArn

The role ARN for the group.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:([\w+=/,.@-]*)?:[0-9]+:[\w+=/,.@-]+(:[\w+=/,.@-]+)?(:[\w+=/,.@-]+)?`

Required: No

UserPoolId

The user pool ID for the user pool.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

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

HttpHeader

The HTTP header.

Contents

headerName

The header name

Type: String

Required: No

headerValue

The header value.

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

IdentityProviderType

A container for information about an identity provider.

Contents

AttributeMapping

A mapping of identity provider attributes to standard and custom user pool attributes.

Type: String to string map

Key Length Constraints: Minimum length of 1. Maximum length of 32.

Required: No

CreationDate

The date the identity provider was created.

Type: Timestamp

Required: No

IdpIdentifiers

A list of identity provider identifiers.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 50 items.

Length Constraints: Minimum length of 1. Maximum length of 40.

Pattern: `[\w\s+=.@-]+`

Required: No

LastModifiedDate

The date the identity provider was last modified.

Type: Timestamp

Required: No

ProviderDetails

The identity provider details, such as `MetadataURL` and `MetadataFile`.

Type: String to string map

Required: No

ProviderName

The identity provider name.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 32.

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

Required: No

ProviderType

The identity provider type.

Type: String

Valid Values: `SAML | Facebook | Google | LoginWithAmazon | OIDC`

Required: No

UserPoolId

The user pool ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

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

LambdaConfigType

Specifies the configuration for AWS Lambda triggers.

Contents

CreateAuthChallenge

Creates an authentication challenge.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:([\w+=/,.@-]*)?:[0-9]+:[\w+=/,.@-]+(:[\w+=/,.@-]+)?(:[\w+=/,.@-]+)?`

Required: No

CustomMessage

A custom Message AWS Lambda trigger.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:([\w+=/,.@-]*)?:[0-9]+:[\w+=/,.@-]+(:[\w+=/,.@-]+)?(:[\w+=/,.@-]+)?`

Required: No

DefineAuthChallenge

Defines the authentication challenge.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:([\w+=/,.@-]*)?:[0-9]+:[\w+=/,.@-]+(:[\w+=/,.@-]+)?(:[\w+=/,.@-]+)?`

Required: No

PostAuthentication

A post-authentication AWS Lambda trigger.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:([\w+=/,.@-]*)?:[0-9]+:[\w+=/,.@-]+(:[\w+=/,.@-]+)?(:[\w+=/,.@-]+)?`

Required: No

PostConfirmation

A post-confirmation AWS Lambda trigger.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:([\w+=/,.@-]*)?:[0-9]+:[\w+=/,.@-]+(:[\w+=/,.@-]+)?(:[\w+=/,.@-]+)?`

Required: No

PreAuthentication

A pre-authentication AWS Lambda trigger.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:([\w+=/,.@-]*)?:[0-9]+:[\w+=/,.@-]+(:[\w+=/,.@-]+)?(:[\w+=/,.@-]+)?`

Required: No

PreSignUp

A pre-registration AWS Lambda trigger.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:([\w+=/,.@-]*)?:[0-9]+:[\w+=/,.@-]+(:[\w+=/,.@-]+)?(:[\w+=/,.@-]+)?`

Required: No

PreTokenGeneration

A Lambda trigger that is invoked before token generation.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:([\w+=/,.@-]*)?:[0-9]+:[\w+=/,.@-]+(:[\w+=/,.@-]+)?(:[\w+=/,.@-]+)?`

Required: No

UserMigration

The user migration Lambda config type.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:([\w+=/,.@-]*)?:[0-9]+:[\w+=/,.@-]+(:[\w+=/,.@-]+)?(:[\w+=/,.@-]+)?`

Required: No

VerifyAuthChallengeResponse

Verifies the authentication challenge response.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:([\w+=/,.@-]*)?:[0-9]+:[\w+=/,.@-]+(:[\w+=/,.@-]+)?(:[\w+=/,.@-]+)?`

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

MessageTemplateType

The message template structure.

Contents

EmailMessage

The message template for email messages.

Type: String

Length Constraints: Minimum length of 6. Maximum length of 20000.

Pattern: `[\p{L}\p{M}\p{S}\p{N}\p{P}\s*]*\{####\}[\p{L}\p{M}\p{S}\p{N}\p{P}\s*]*`

Required: No

EmailSubject

The subject line for email messages.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 140.

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

Required: No

SMSMessage

The message template for SMS messages.

Type: String

Length Constraints: Minimum length of 6. Maximum length of 140.

Pattern: `.*\{####\}.*`

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

MFAOptionType

Specifies the different settings for multi-factor authentication (MFA).

Contents

AttributeName

The attribute name of the MFA option type.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 32.

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

Required: No

DeliveryMedium

The delivery medium (email message or SMS message) to send the MFA code.

Type: String

Valid Values: `SMS | EMAIL`

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

NewDeviceMetadataType

The new device metadata type.

Contents

DeviceGroupKey

The device group key.

Type: String

Required: No

DeviceKey

The device key.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-f-]+`

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

NotifyConfigurationType

The notify configuration type.

Contents

BlockEmail

Email template used when a detected risk event is blocked.

Type: NotifyEmailType (p. 348) object

Required: No

From

The email address that is sending the email. It must be either individually verified with Amazon SES, or from a domain that has been verified with Amazon SES.

Type: String

Required: No

MfaEmail

The MFA email template used when MFA is challenged as part of a detected risk.

Type: NotifyEmailType (p. 348) object

Required: No

NoActionEmail

The email template used when a detected risk event is allowed.

Type: NotifyEmailType (p. 348) object

Required: No

ReplyTo

The destination to which the receiver of an email should reply to.

Type: String

Required: No

SourceArn

The Amazon Resource Name (ARN) of the identity that is associated with the sending authorization policy. It permits Amazon Cognito to send for the email address specified in the `From` parameter.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:([\w+=/,.@-]*)?:[0-9]+:[\w+=/,.@-]+(:[\w+=/,.@-]+)?(:[\w+=/,.@-]+)?`

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

NotifyEmailType

The notify email type.

Contents

HtmlBody

The HTML body.

Type: String

Length Constraints: Minimum length of 6. Maximum length of 20000.

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

Required: No

Subject

The subject.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 140.

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

Required: Yes

TextBody

The text body.

Type: String

Length Constraints: Minimum length of 6. Maximum length of 20000.

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

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

NumberAttributeConstraintsType

The minimum and maximum value of an attribute that is of the number data type.

Contents

MaxValue

The maximum value of an attribute that is of the number data type.

Type: String

Required: No

MinValue

The minimum value of an attribute that is of the number data type.

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

PasswordPolicyType

The password policy type.

Contents

MinimumLength

The minimum length of the password policy that you have set. Cannot be less than 6.

Type: Integer

Valid Range: Minimum value of 6. Maximum value of 99.

Required: No

RequireLowercase

In the password policy that you have set, refers to whether you have required users to use at least one lowercase letter in their password.

Type: Boolean

Required: No

RequireNumbers

In the password policy that you have set, refers to whether you have required users to use at least one number in their password.

Type: Boolean

Required: No

RequireSymbols

In the password policy that you have set, refers to whether you have required users to use at least one symbol in their password.

Type: Boolean

Required: No

RequireUppercase

In the password policy that you have set, refers to whether you have required users to use at least one uppercase letter in their password.

Type: Boolean

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

ProviderDescription

A container for identity provider details.

Contents

CreationDate

The date the provider was added to the user pool.

Type: Timestamp

Required: No

LastModifiedDate

The date the provider was last modified.

Type: Timestamp

Required: No

ProviderName

The identity provider name.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 32.

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

Required: No

ProviderType

The identity provider type.

Type: String

Valid Values: `SAML | Facebook | Google | LoginWithAmazon | OIDC`

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

ProviderUserIdentifierType

A container for information about an identity provider for a user pool.

Contents

ProviderAttributeName

The name of the provider attribute to link to, for example, `NameID`.

Type: String

Required: No

ProviderAttributeValue

The value of the provider attribute to link to, for example, `xxxxx_account`.

Type: String

Required: No

ProviderName

The name of the provider, for example, Facebook, Google, or Login with Amazon.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 32.

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

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

ResourceServerScopeType

A resource server scope.

Contents

ScopeDescription

A description of the scope.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: Yes

ScopeName

The name of the scope.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `[\x21\x23-\x2E\x30-\x5B\x5D-\x7E]+`

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

ResourceServerType

A container for information about a resource server for a user pool.

Contents

Identifier

The identifier for the resource server.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `[\x21\x23-\x5B\x5D-\x7E]+`

Required: No

Name

The name of the resource server.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `[\w\s+=,.@-]+`

Required: No

Scopes

A list of scopes that are defined for the resource server.

Type: Array of ResourceServerScopeType (p. 354) objects

Array Members: Maximum number of 25 items.

Required: No

UserPoolId

The user pool ID for the user pool that hosts the resource server.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

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

RiskConfigurationType

The risk configuration type.

Contents

AccountTakeoverRiskConfiguration

The account takeover risk configuration object including the `NotifyConfiguration` object and `Actions` to take in the case of an account takeover.

Type: AccountTakeoverRiskConfigurationType (p. 311) object

Required: No

ClientId

The app client ID.

Type: String

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

Pattern: `[\w+]+`

Required: No

CompromisedCredentialsRiskConfiguration

The compromised credentials risk configuration object including the `EventFilter` and the `EventAction`

Type: CompromisedCredentialsRiskConfigurationType (p. 323) object

Required: No

LastModifiedDate

The last modified date.

Type: Timestamp

Required: No

RiskExceptionConfiguration

The configuration to override the risk decision.

Type: RiskExceptionConfigurationType (p. 359) object

Required: No

UserPoolId

The user pool ID.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

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

RiskExceptionConfigurationType

The type of the configuration to override the risk decision.

Contents

BlockedIPRangeList

Overrides the risk decision to always block the pre-authentication requests. The IP range is in CIDR notation: a compact representation of an IP address and its associated routing prefix.

Type: Array of strings

Array Members: Maximum number of 20 items.

Required: No

SkippedIPRangeList

Risk detection is not performed on the IP addresses in the range list. The IP range is in CIDR notation.

Type: Array of strings

Array Members: Maximum number of 20 items.

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

SchemaAttributeType

Contains information about the schema attribute.

Contents

AttributeDataType

The attribute data type.

Type: String

Valid Values: `String | Number | DateTime | Boolean`

Required: No

DeveloperOnlyAttribute

Specifies whether the attribute type is developer only.

Type: Boolean

Required: No

Mutable

Specifies whether the value of the attribute can be changed.

For any user pool attribute that's mapped to an identity provider attribute, you must set this parameter to `true`. Amazon Cognito updates mapped attributes when users sign in to your application through an identity provider. If an attribute is immutable, Amazon Cognito throws an error when it attempts to update the attribute. For more information, see Specifying Identity Provider Attribute Mappings for Your User Pool.

Type: Boolean

Required: No

Name

A schema attribute of the name type.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 20.

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

Required: No

NumberAttributeConstraints

Specifies the constraints for an attribute of the number type.

Type: NumberAttributeConstraintsType (p. 349) object

Required: No

Required

Specifies whether a user pool attribute is required. If the attribute is required and the user does not provide a value, registration or sign-in will fail.

Type: Boolean

Required: No

StringAttributeConstraints

Specifies the constraints for an attribute of the string type.

Type: StringAttributeConstraintsType (p. 367) object

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

SmsConfigurationType

The SMS configuration type.

Contents

ExternalId

The external ID.

Type: String

Required: No

SnsCallerArn

The Amazon Resource Name (ARN) of the Amazon Simple Notification Service (SNS) caller.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:([\w+=/,.@-]*)?:[0-9]+:[\w+=/,.@-]+(:[\w+=/,.@-]+)?(:[\w+=/,.@-]+)?`

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

SmsMfaConfigType

The SMS text message multi-factor authentication (MFA) configuration type.

Contents

SmsAuthenticationMessage

The SMS authentication message.

Type: String

Length Constraints: Minimum length of 6. Maximum length of 140.

Pattern: `.*\{####\}.*`

Required: No

SmsConfiguration

The SMS configuration.

Type: SmsConfigurationType (p. 362) object

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

SMSMfaSettingsType

The SMS multi-factor authentication (MFA) settings type.

Contents

Enabled

Specifies whether SMS text message MFA is enabled.

Type: Boolean

Required: No

PreferredMfa

The preferred MFA method.

Type: Boolean

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

SoftwareTokenMfaConfigType

The type used for enabling software token MFA at the user pool level.

Contents

Enabled

Specifies whether software token MFA is enabled.

Type: Boolean

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

SoftwareTokenMfaSettingsType

The type used for enabling software token MFA at the user level.

Contents

Enabled

Specifies whether software token MFA is enabled.

Type: Boolean

Required: No

PreferredMfa

The preferred MFA method.

Type: Boolean

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

StringAttributeConstraintsType

The constraints associated with a string attribute.

Contents

MaxLength

The maximum length.

Type: String

Required: No

MinLength

The minimum length.

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

UICustomizationType

A container for the UI customization information for a user pool's built-in app UI.

Contents

ClientId

The client ID for the client app.

Type: String

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

Pattern: `[\w+]+`

Required: No

CreationDate

The creation date for the UI customization.

Type: Timestamp

Required: No

CSS

The CSS values in the UI customization.

Type: String

Required: No

CSSVersion

The CSS version number.

Type: String

Required: No

ImageUrl

The logo image for the UI customization.

Type: String

Required: No

LastModifiedDate

The last-modified date for the UI customization.

Type: Timestamp

Required: No

UserPoolId

The user pool ID for the user pool.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

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

UserContextDataType

Contextual data such as the user's device fingerprint, IP address, or location used for evaluating the risk of an unexpected event by Amazon Cognito advanced security.

Contents

EncodedData

Contextual data such as the user's device fingerprint, IP address, or location used for evaluating the risk of an unexpected event by Amazon Cognito advanced security.

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

UserImportJobType

The user import job type.

Contents

CloudWatchLogsRoleArn

The role ARN for the Amazon CloudWatch Logging role for the user import job. For more information, see "Creating the CloudWatch Logs IAM Role" in the Amazon Cognito Developer Guide.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:([\w+=/,.@-]*)?:[0-9]+:[\w+=/,.@-]+(:[\w+=/,.@-]+)?(:[\w+=/,.@-]+)?`

Required: No

CompletionDate

The date when the user import job was completed.

Type: Timestamp

Required: No

CompletionMessage

The message returned when the user import job is completed.

Type: String

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

Pattern: `[\w]+`

Required: No

CreationDate

The date the user import job was created.

Type: Timestamp

Required: No

FailedUsers

The number of users that could not be imported.

Type: Long

Required: No

ImportedUsers

The number of users that were successfully imported.

Type: Long

Required: No

JobId

The job ID for the user import job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `import-[0-9a-zA-Z-]+`

Required: No

JobName

The job name for the user import job.

Type: String

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

Pattern: `[\w\s+=,.@-]+`

Required: No

PreSignedUrl

The pre-signed URL to be used to upload the `.csv` file.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 2048.

Required: No

SkippedUsers

The number of users that were skipped.

Type: Long

Required: No

StartDate

The date when the user import job was started.

Type: Timestamp

Required: No

Status

The status of the user import job. One of the following:

- `Created` - The job was created but not started.
- `Pending` - A transition state. You have started the job, but it has not begun importing users yet.
- `InProgress` - The job has started, and users are being imported.
- `Stopping` - You have stopped the job, but the job has not stopped importing users yet.
- `Stopped` - You have stopped the job, and the job has stopped importing users.
- `Succeeded` - The job has completed successfully.
- `Failed` - The job has stopped due to an error.
- `Expired` - You created a job, but did not start the job within 24-48 hours. All data associated with the job was deleted, and the job cannot be started.

Type: String

Valid Values: `Created | Pending | InProgress | Stopping | Expired | Stopped | Failed | Succeeded`

Required: No

UserPoolId

The user pool ID for the user pool that the users are being imported into.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

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

UserPoolAddOnsType

The user pool add-ons type.

Contents

AdvancedSecurityMode

The advanced security mode.

Type: String

Valid Values: `OFF | AUDIT | ENFORCED`

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

UserPoolClientDescription

The description of the user pool client.

Contents

ClientId

The ID of the client associated with the user pool.

Type: String

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

Pattern: `[\w+]+`

Required: No

ClientName

The client name from the user pool client description.

Type: String

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

Pattern: `[\w\s+=,.@-]+`

Required: No

UserPoolId

The user pool ID for the user pool where you want to describe the user pool client.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

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

UserPoolClientType

Contains information about a user pool client.

Contents

AllowedOAuthFlows

Set to `code` to initiate a code grant flow, which provides an authorization code as the response. This code can be exchanged for access tokens with the token endpoint.

Set to `token` to specify that the client should get the access token (and, optionally, ID token, based on scopes) directly.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 3 items.

Valid Values: `code | implicit | client_credentials`

Required: No

AllowedOAuthFlowsUserPoolClient

Set to TRUE if the client is allowed to follow the OAuth protocol when interacting with Cognito user pools.

Type: Boolean

Required: No

AllowedOAuthScopes

A list of allowed `OAuth` scopes. Currently supported values are `"phone"`, `"email"`, `"openid"`, and `"Cognito"`.

Type: Array of strings

Array Members: Maximum number of 25 items.

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `[\x21\x23-\x5B\x5D-\x7E]+`

Required: No

AnalyticsConfiguration

The Amazon Pinpoint analytics configuration for the user pool client.

Type: AnalyticsConfigurationType (p. 313) object

Required: No

CallbackURLs

A list of allowed redirect (callback) URLs for the identity providers.

A redirect URI must:

- Be an absolute URI.
- Be registered with the authorization server.

- Not include a fragment component.

See OAuth 2.0 - Redirection Endpoint.

Amazon Cognito requires HTTPS over HTTP except for http://localhost for testing purposes only.

App callback URLs such as myapp://example are also supported.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 100 items.

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

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

Required: No

ClientId

The ID of the client associated with the user pool.

Type: String

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

Pattern: `[\w+]+`

Required: No

ClientName

The client name from the user pool request of the client type.

Type: String

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

Pattern: `[\w\s+=,.@-]+`

Required: No

ClientSecret

The client secret from the user pool request of the client type.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `[\w+]+`

Required: No

CreationDate

The date the user pool client was created.

Type: Timestamp

Required: No

DefaultRedirectURI

The default redirect URI. Must be in the `CallbackURLs` list.

A redirect URI must:

- Be an absolute URI.
- Be registered with the authorization server.
- Not include a fragment component.

See OAuth 2.0 - Redirection Endpoint.

Amazon Cognito requires HTTPS over HTTP except for http://localhost for testing purposes only.

App callback URLs such as myapp://example are also supported.

Type: String

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

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

Required: No

ExplicitAuthFlows

The explicit authentication flows.

Type: Array of strings

Valid Values: `ADMIN_NO_SRP_AUTH | CUSTOM_AUTH_FLOW_ONLY | USER_PASSWORD_AUTH`

Required: No

LastModifiedDate

The date the user pool client was last modified.

Type: Timestamp

Required: No

LogoutURLs

A list of allowed logout URLs for the identity providers.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 100 items.

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

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

Required: No

ReadAttributes

The Read-only attributes.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: No

RefreshTokenValidity

The time limit, in days, after which the refresh token is no longer valid and cannot be used.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 3650.

Required: No

SupportedIdentityProviders

A list of provider names for the identity providers that are supported on this client.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 32.

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

Required: No

UserPoolId

The user pool ID for the user pool client.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: No

WriteAttributes

The writeable attributes.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 2048.

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

UserPoolDescriptionType

A user pool description.

Contents

CreationDate

The date the user pool description was created.

Type: Timestamp

Required: No

Id

The ID in a user pool description.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: No

LambdaConfig

The AWS Lambda configuration information in a user pool description.

Type: LambdaConfigType (p. 340) object

Required: No

LastModifiedDate

The date the user pool description was last modified.

Type: Timestamp

Required: No

Name

The name in a user pool description.

Type: String

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

Pattern: `[\w\s+=,.@-]+`

Required: No

Status

The user pool status in a user pool description.

Type: String

Valid Values: `Enabled | Disabled`

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

UserPoolPolicyType

The policy associated with a user pool.

Contents

PasswordPolicy

The password policy.

Type: PasswordPolicyType (p. 350) object

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

UserPoolType

A container for information about the user pool.

Contents

AdminCreateUserConfig

The configuration for `AdminCreateUser` requests.

Type: AdminCreateUserConfigType (p. 312) object

Required: No

AliasAttributes

Specifies the attributes that are aliased in a user pool.

Type: Array of strings

Valid Values: `phone_number | email | preferred_username`

Required: No

Arn

The Amazon Resource Name (ARN) for the user pool.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Pattern: `arn:[\w+=/,.@-]+:[\w+=/,.@-]+:([\w+=/,.@-]*)?:[0-9]+:[\w+=/,.@-]+(:[\w+=/,.@-]+)?(:[\w+=/,.@-]+)?`

Required: No

AutoVerifiedAttributes

Specifies the attributes that are auto-verified in a user pool.

Type: Array of strings

Valid Values: `phone_number | email`

Required: No

CreationDate

The date the user pool was created.

Type: Timestamp

Required: No

CustomDomain

Type: String

Length Constraints: Minimum length of 1. Maximum length of 63.

Pattern: `^[a-z0-9](?:[a-z0-9\-]{0,61}[a-z0-9])?$`

Required: No

DeviceConfiguration

The device configuration.

Type: DeviceConfigurationType (p. 326) object

Required: No

Domain

Holds the domain prefix if the user pool has a domain associated with it.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 63.

Pattern: `^[a-z0-9](?:[a-z0-9\-]{0,61}[a-z0-9])?$`

Required: No

EmailConfiguration

The email configuration.

Type: EmailConfigurationType (p. 331) object

Required: No

EmailConfigurationFailure

The reason why the email configuration cannot send the messages to your users.

Type: String

Required: No

EmailVerificationMessage

The contents of the email verification message.

Type: String

Length Constraints: Minimum length of 6. Maximum length of 20000.

Pattern: `[\p{L}\p{M}\p{S}\p{N}\p{P}\s*]*\{####\}[\p{L}\p{M}\p{S}\p{N}\p{P}\s*]*`

Required: No

EmailVerificationSubject

The subject of the email verification message.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 140.

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

Required: No

EstimatedNumberOfUsers

A number estimating the size of the user pool.

Type: Integer

Required: No

Id

The ID of the user pool.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 55.

Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: No

LambdaConfig

The AWS Lambda triggers associated with the user pool.

Type: LambdaConfigType (p. 340) object

Required: No

LastModifiedDate

The date the user pool was last modified.

Type: Timestamp

Required: No

MfaConfiguration

Can be one of the following values:

- `OFF` - MFA tokens are not required and cannot be specified during user registration.
- `ON` - MFA tokens are required for all user registrations. You can only specify required when you are initially creating a user pool.
- `OPTIONAL` - Users have the option when registering to create an MFA token.

Type: String

Valid Values: `OFF | ON | OPTIONAL`

Required: No

Name

The name of the user pool.

Type: String

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

Pattern: `[\w\s+=,.@-]+`

Required: No

Policies

The policies associated with the user pool.

Type: UserPoolPolicyType (p. 382) object

Required: No

SchemaAttributes

A container with the schema attributes of a user pool.

Type: Array of SchemaAttributeType (p. 360) objects

Array Members: Minimum number of 1 item. Maximum number of 50 items.

Required: No

SmsAuthenticationMessage

The contents of the SMS authentication message.

Type: String

Length Constraints: Minimum length of 6. Maximum length of 140.

Pattern: `.*\{####\}.*`

Required: No

SmsConfiguration

The SMS configuration.

Type: SmsConfigurationType (p. 362) object

Required: No

SmsConfigurationFailure

The reason why the SMS configuration cannot send the messages to your users.

Type: String

Required: No

SmsVerificationMessage

The contents of the SMS verification message.

Type: String

Length Constraints: Minimum length of 6. Maximum length of 140.

Pattern: `.*\{####\}.*`

Required: No

Status

The status of a user pool.

Type: String

Valid Values: `Enabled | Disabled`

Required: No

UsernameAttributes

Specifies whether email addresses or phone numbers can be specified as usernames when a user signs up.

Type: Array of strings

Valid Values: `phone_number | email`

Required: No

UserPoolAddOns

The user pool add-ons.

Type: UserPoolAddOnsType (p. 374) object

Required: No

UserPoolTags

The cost allocation tags for the user pool. For more information, see Adding Cost Allocation Tags to Your User Pool

Type: String to string map

Required: No

VerificationMessageTemplate

The template for verification messages.

Type: VerificationMessageTemplateType (p. 390) object

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

UserType

The user type.

Contents

Attributes

A container with information about the user type attributes.

Type: Array of AttributeType (p. 315) objects

Required: No

Enabled

Specifies whether the user is enabled.

Type: Boolean

Required: No

MFAOptions

The MFA options for the user.

Type: Array of MFAOptionType (p. 344) objects

Required: No

UserCreateDate

The creation date of the user.

Type: Timestamp

Required: No

UserLastModifiedDate

The last modified date of the user.

Type: Timestamp

Required: No

Username

The user name of the user you wish to describe.

Type: String

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

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

Required: No

UserStatus

The user status. Can be one of the following:

- UNCONFIRMED - User has been created but not confirmed.
- CONFIRMED - User has been confirmed.

- ARCHIVED - User is no longer active.
- COMPROMISED - User is disabled due to a potential security threat.
- UNKNOWN - User status is not known.

Type: String

Valid Values: `UNCONFIRMED | CONFIRMED | ARCHIVED | COMPROMISED | UNKNOWN | RESET_REQUIRED | FORCE_CHANGE_PASSWORD`

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

VerificationMessageTemplateType

The template for verification messages.

Contents

DefaultEmailOption

The default email option.

Type: String

Valid Values: `CONFIRM_WITH_LINK | CONFIRM_WITH_CODE`

Required: No

EmailMessage

The email message template.

Type: String

Length Constraints: Minimum length of 6. Maximum length of 20000.

Pattern: `[\p{L}\p{M}\p{S}\p{N}\p{P}\s*]*\{####\}[\p{L}\p{M}\p{S}\p{N}\p{P}\s*]*`

Required: No

EmailMessageByLink

The email message template for sending a confirmation link to the user.

Type: String

Length Constraints: Minimum length of 6. Maximum length of 20000.

Pattern: `[\p{L}\p{M}\p{S}\p{N}\p{P}\s*]*\{##[\p{L}\p{M}\p{S}\p{N}\p{P}\s*]*##\}[\p{L}\p{M}\p{S}\p{N}\p{P}\s*]*`

Required: No

EmailSubject

The subject line for the email message template.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 140.

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

Required: No

EmailSubjectByLink

The subject line for the email message template for sending a confirmation link to the user.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 140.

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

Required: No

SmsMessage

The SMS message template.

Type: String

Length Constraints: Minimum length of 6. Maximum length of 140.

Pattern: `.*\{####\}.*`

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