Amazon Cognito

API Reference

API Version 2016-04-18

Amazon Cognito: 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 June 6, 2022.

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. 450).

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. 410) 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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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're 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

AdminAddUserToGroup

Adds the specified user to the specified group.

Calling this action 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. 450).

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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

AdminConfirmSignUp

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

Calling this action requires developer credentials.

Request Syntax

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

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 450).

The request accepts the following data in JSON format.

ClientMetadata (p. 9)

A map of custom key-value pairs that you can provide as input for any custom workflows that this action triggers.

If your user pool configuration includes triggers, the AdminConfirmSignUp API action invokes the AWS Lambda function that is specified for the *post confirmation* trigger. When Amazon Cognito invokes this function, it passes a JSON payload, which the function receives as input. In this payload, the `clientMetadata` attribute provides the data that you assigned to the ClientMetadata parameter in your AdminConfirmSignUp request. In your function code in Lambda, you can process the ClientMetadata value to enhance your workflow for your specific needs.

For more information, see Customizing user pool Workflows with Lambda Triggers in the *Amazon Cognito Developer Guide*.

Note

When you use the ClientMetadata parameter, remember that Amazon Cognito won't do the following:

- Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn't include triggers, the ClientMetadata parameter serves no purpose.
- Validate the ClientMetadata value.
- Encrypt the ClientMetadata value. Don't use Amazon Cognito to provide sensitive information.

Type: String to string map

Required: No

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. 452).

InternalErrorException

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

HTTP Status Code: 500

InvalidLambdaResponseException

This exception is thrown when Amazon Cognito 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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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, such as 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 Amazon Cognito encounters an unexpected exception with AWS Lambda.

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 isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

AdminCreateUser

Creates a new user in the specified user pool.

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

> **Note**
> This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.
> If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In *sandbox mode*, you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see SMS message settings for Amazon Cognito user pools in the *Amazon Cognito Developer Guide*.

This message is based on a template that you configured in your call to create or update a user pool. 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 won't 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

```
{
   "ClientMetadata": { 
      "string" : "string" 
   },
   "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. 450).

The request accepts the following data in JSON format.

ClientMetadata (p. 12)

A map of custom key-value pairs that you can provide as input for any custom workflows that this action triggers.

You create custom workflows by assigning AWS Lambda functions to user pool triggers. When you use the AdminCreateUser API action, Amazon Cognito invokes the function that is assigned to the *pre sign-up* trigger. When Amazon Cognito invokes this function, it passes a JSON payload, which the function receives as input. This payload contains a `clientMetadata` attribute, which provides the data that you assigned to the ClientMetadata parameter in your AdminCreateUser request. In your function code in AWS Lambda, you can process the `clientMetadata` value to enhance your workflow for your specific needs.

For more information, see Customizing user pool Workflows with Lambda Triggers in the *Amazon Cognito Developer Guide*.

Note

When you use the ClientMetadata parameter, remember that Amazon Cognito won't do the following:

- Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn't include triggers, the ClientMetadata parameter serves no purpose.
- Validate the ClientMetadata value.
- Encrypt the ClientMetadata value. Don't use Amazon Cognito to provide sensitive information.

Type: String to string map

Required: No

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"`. You can specify more than one value.

Type: Array of strings

Valid Values: `SMS | EMAIL`

Required: No

ForceAliasCreation (p. 12)

This parameter is used only 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. You can specify only one value.

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 isn't required. If you don't 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: 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 (when creating a user pool or in the **Attributes** tab of the console) either you should supply (in your call to `AdminCreateUser`) or the user should supply (when they sign 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. You can do this 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.

- **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. 359) 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 can't 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 AWS 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 isn't persisted.

Type: Array of AttributeType (p. 359) 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. 15)

The newly created user.

Type: UserType (p. 446) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 Amazon Cognito 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 Amazon Cognito encounters an invalid password.

HTTP Status Code: 400

InvalidSmsRoleAccessPolicyException

This exception is returned when the role provided for SMS configuration doesn't have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't 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 isn't 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 can't 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 Amazon Cognito encounters an unexpected exception with AWS Lambda.

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 isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

AdminDeleteUser

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

Calling this action 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. 450).

The request accepts the following data in JSON format.

Username (p. 19)

The user name of the user you want 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. 19)

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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

AdminDeleteUserAttributes

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

Calling this action 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. 450).

The request accepts the following data in JSON format.

UserAttributeNames (p. 21)

An array of strings representing the user attribute names you want 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. 21)

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. 21)

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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

AdminDisableProviderForUser

Prevents the user from signing in with the specified external (SAML or social) identity provider (IdP). If the user that you want to deactivate is a Amazon Cognito user pools native username + password user, they can't use their password to sign in. If the user to deactivate is a linked external IdP user, any link between that user and an existing user is removed. When the external user signs in again, and the user is no longer attached to the previously linked `DestinationUser`, the user must create a new user account. See AdminLinkProviderForUser.

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

The `ProviderAttributeName` must always be `Cognito_Subject` for social IdPs. 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 using `AdminLinkProviderForUser` 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. 450).

The request accepts the following data in JSON format.

User (p. 23)

The user to be disabled.

Type: ProviderUserIdentifierType (p. 402) object

Required: Yes

UserPoolId (p. 23)

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. 452).

AliasExistsException

This exception is thrown when a user tries to confirm the account with an email address or phone number that has already been supplied as an alias for a different user profile. This exception indicates that an account with this email address or phone already exists in a user pool that you've configured to use email address or phone number as a sign-in alias.

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

AdminDisableUser

Disables the specified user.

Calling this action 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. 450).

The request accepts the following data in JSON format.

Username (p. 26)

The user name of the user you want 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. 26)

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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

AdminEnableUser

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

Calling this action 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. 450).

The request accepts the following data in JSON format.

Username (p. 28)

The user name of the user you want 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. 28)

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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

AdminForgetDevice

Forgets the device, as an administrator.

Calling this action 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. 450).

The request accepts the following data in JSON format.

DeviceKey (p. 30)

The device key.

Type: String

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

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

Required: Yes

Username (p. 30)

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. 30)

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. 452).

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 not valid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

AdminGetDevice

Gets the device, as an administrator.

Calling this action 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. 450).

The request accepts the following data in JSON format.

DeviceKey (p. 32)

The device key.

Type: String

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

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

Required: Yes

Username (p. 32)

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. 32)

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. 32)

The device.

Type: DeviceType (p. 374) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 not valid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

AdminGetUser

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

Calling this action 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. 450).

The request accepts the following data in JSON format.

Username (p. 35)

The user name of the user you want 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. 35)

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. 35)

Indicates that the status is `enabled`.

Type: Boolean

MFAOptions (p. 35)

This response parameter is no longer supported. It provides information only about SMS MFA configurations. It doesn't provide information about time-based one-time password (TOTP) software token MFA configurations. To look up information about either type of MFA configuration, use UserMFASettingList instead.

Type: Array of MFAOptionType (p. 393) objects

PreferredMfaSetting (p. 35)

The user's preferred MFA setting.

Type: String

UserAttributes (p. 35)

An array of name-value pairs representing user attributes.

Type: Array of AttributeType (p. 359) objects

UserCreateDate (p. 35)

The date the user was created.

Type: Timestamp

UserLastModifiedDate (p. 35)

The date the user was last modified.

Type: Timestamp

UserMFASettingList (p. 35)

The MFA options that are activated for the user. The possible values in this list are `SMS_MFA` and `SOFTWARE_TOKEN_MFA`.

Type: Array of strings

Username (p. 35)

The user name of the user about whom you're 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. 35)

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.
- UNKNOWN - User status isn't known.
- RESET_REQUIRED - User is confirmed, but the user must request a code and reset their password before they can sign in.
- FORCE_CHANGE_PASSWORD - The user is confirmed and the user can sign in using a temporary password, but on first sign-in, the user must change their password to a new value before doing anything else.

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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

AdminInitiateAuth

Initiates the authentication flow, as an administrator.

> **Note**
> This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.
> If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In *sandbox mode* , you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see SMS message settings for Amazon Cognito user pools in the *Amazon Cognito Developer Guide*.

Calling this action 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. 450).

The request accepts the following data in JSON format.

AnalyticsMetadata (p. 39)

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

Type: AnalyticsMetadataType (p. 358) object

Required: No

AuthFlow (p. 39)

The authentication flow for this call to run. 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 Secure Remote Password (SRP) protocol variables to be used for next challenge execution.
- `ADMIN_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.
- `ADMIN_USER_PASSWORD_AUTH`: Admin-based user password authentication. This replaces the `ADMIN_NO_SRP_AUTH` authentication flow. In this flow, Amazon Cognito receives the password in the request instead of using the SRP process to verify passwords.

Type: String

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

Required: Yes

AuthParameters (p. 39)

The authentication parameters. These are inputs corresponding to the `AuthFlow` that you're 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`. To start the authentication flow with password verification, include `ChallengeName: SRP_A` and `SRP_A: (The SRP_A Value)`.

Type: String to string map

Required: No

ClientId (p. 39)

The app client ID.

Type: String

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

Pattern: `[\w+]+`

Required: Yes

ClientMetadata (p. 39)

A map of custom key-value pairs that you can provide as input for certain custom workflows that this action triggers.

You create custom workflows by assigning AWS Lambda functions to user pool triggers. When you use the AdminInitiateAuth API action, Amazon Cognito invokes the Lambda functions that are specified for various triggers. The ClientMetadata value is passed as input to the functions for only the following triggers:

- Pre signup
- Pre authentication
- User migration

When Amazon Cognito invokes the functions for these triggers, it passes a JSON payload, which the function receives as input. This payload contains a `validationData` attribute, which provides the data that you assigned to the ClientMetadata parameter in your AdminInitiateAuth request. In your function code in AWS Lambda, you can process the `validationData` value to enhance your workflow for your specific needs.

When you use the AdminInitiateAuth API action, Amazon Cognito also invokes the functions for the following triggers, but it doesn't provide the ClientMetadata value as input:

- Post authentication
- Custom message
- Pre token generation
- Create auth challenge
- Define auth challenge
- Verify auth challenge

For more information, see Customizing user pool Workflows with Lambda Triggers in the *Amazon Cognito Developer Guide*.

> **Note**
> When you use the ClientMetadata parameter, remember that Amazon Cognito won't do the following:
>
> - Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn't include triggers, the ClientMetadata parameter serves no purpose.
> - Validate the ClientMetadata value.
> - Encrypt the ClientMetadata value. Don't use Amazon Cognito to provide sensitive information.

Type: String to string map

Required: No

ContextData (p. 39)

Contextual data about your user session, such as the device fingerprint, IP address, or location. Amazon Cognito advanced security evaluates the risk of an authentication event based on the context that your app generates and passes to Amazon Cognito when it makes API requests.

Type: ContextDataType (p. 368) object

Required: No

UserPoolId (p. 39)

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. 42)

The result of the authentication response. This is only returned if the caller doesn't 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. 360) object

ChallengeName (p. 42)

The name of the challenge that you're responding to with this call. This is returned in the `AdminInitiateAuth` response if you must pass another challenge.

- `MFA_SETUP`: If MFA is required, users who don't 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 time-based one-time password (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 activated in 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 must authenticate with USERNAME and PASSWORD directly. An app client must be enabled to use this flow.
- NEW_PASSWORD_REQUIRED: For users who are required to change their passwords after successful first login. Respond to this challenge with NEW_PASSWORD and any required attributes that Amazon Cognito returned in the requiredAttributes parameter. You can also set values for attributes that aren't required by your user pool and that your app client can write. For more information, see AdminRespondToAuthChallenge.

  Note
  In a NEW_PASSWORD_REQUIRED challenge response, you can't modify a required attribute that already has a value. In AdminRespondToAuthChallenge, set a value for any keys that Amazon Cognito returned in the requiredAttributes parameter, then use the AdminUpdateUserAttributes API operation to modify the value of any additional attributes.
- MFA_SETUP: For users who are required to set up an MFA factor before they can sign in. The MFA types activated for the user pool will be listed in the challenge parameters MFA_CAN_SETUP value.

  To set up software token MFA, use the session returned here from InitiateAuth as an input to AssociateSoftwareToken, and use the session returned by VerifySoftwareToken as an input to RespondToAuthChallenge with challenge name MFA_SETUP to complete sign-in. To set up SMS MFA, users will need help from an administrator to add a phone number to their account and then call InitiateAuth again to restart sign-in.

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. 42)

The challenge parameters. These are returned to you in the AdminInitiateAuth response if you must 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 is 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 happens because, in the AdminRespondToAuthChallenge API ChallengeResponses, the USERNAME attribute can't be an alias.

Type: String to string map

Session (p. 42)

The session that should be passed both ways in challenge-response calls to the service. If AdminInitiateAuth or AdminRespondToAuthChallenge API call determines that the caller must pass 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. 452).

InternalErrorException

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

HTTP Status Code: 500

InvalidLambdaResponseException

This exception is thrown when Amazon Cognito 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 doesn't have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't 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 not valid.

HTTP Status Code: 400

MFAMethodNotFoundException

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

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't 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 can't 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 Amazon Cognito encounters an unexpected exception with AWS Lambda.

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 isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

AdminLinkProviderForUser

Links an existing user account in a user pool (`DestinationUser`) to an identity from an external IdP (`SourceUser`) based on a specified attribute name and value from the external IdP. 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. You can then use the federated user identity 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. When the user signs in with a federated user identity, they sign in as the existing user account.

> **Note**
> The maximum number of federated identities linked to a user is five.

> **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 IdPs and provider attributes that have been trusted by the application owner.

See also AdminDisableProviderForUser (p. 23).

This action is administrative 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. 450).

The request accepts the following data in JSON format.

DestinationUser (p. 46)

The existing user in the user pool that you want to assign to the external IdP user account. This user can be a native (Username + Password) Amazon Cognito user pools user or a federated user (for example, a SAML or Facebook user). If the user doesn't exist, Amazon Cognito generates an exception. Amazon Cognito returns this user when the new user (with the linked IdP 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.

> **Important**
> All attributes in the DestinationUser profile must be mutable. If you have assigned the user any immutable custom attributes, the operation won't succeed.

Type: ProviderUserIdentifierType (p. 402) object

Required: Yes

SourceUser (p. 46)

An external IdP account for a user who doesn't 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 using a federated social IdP, such as Facebook, Google, or Login with Amazon, you must set the ProviderAttributeName to Cognito_Subject. For social IdPs, the ProviderName will be Facebook, Google, or LoginWithAmazon, and Amazon 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 IdP token.

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

Type: ProviderUserIdentifierType (p. 402) object

Required: Yes

UserPoolId (p. 46)

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. 452).

AliasExistsException

This exception is thrown when a user tries to confirm the account with an email address or phone number that has already been supplied as an alias for a different user profile. This exception indicates that an account with this email address or phone already exists in a user pool that you've configured to use email address or phone number as a sign-in alias.

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

AdminListDevices

Lists devices, as an administrator.

Calling this action 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. 450).

The request accepts the following data in JSON format.

Limit (p. 49)

The limit of the devices request.

Type: Integer

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

Required: No

PaginationToken (p. 49)

The pagination token.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

Required: No

Username (p. 49)

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. 49)

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. 50)

The devices in the list of devices response.

Type: Array of DeviceType (p. 374) objects

PaginationToken (p. 50)

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. 452).

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 not valid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

AdminListGroupsForUser

Lists the groups that the user belongs to.

Calling this action 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. 450).

The request accepts the following data in JSON format.

Limit (p. 52)

The limit of the request to list groups.

Type: Integer

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

Required: No

NextToken (p. 52)

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. 52)

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. 52)

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. 53)

The groups that the user belongs to.

Type: Array of GroupType (p. 383) objects

NextToken (p. 53)

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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

AdminListUserAuthEvents

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. 450).

The request accepts the following data in JSON format.

MaxResults (p. 55)

The maximum number of authentication events to return.

Type: Integer

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

Required: No

NextToken (p. 55)

A pagination token.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

Required: No

Username (p. 55)

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. 55)

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": {
            "CompromisedCredentialsDetected": boolean,
            "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. 56)

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

Type: Array of AuthEventType (p. 362) objects

NextToken (p. 56)

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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 isn't found.

HTTP Status Code: 400

UserPoolAddOnNotEnabledException

This exception is thrown when user pool add-ons aren't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python

- AWS SDK for Ruby V3

AdminRemoveUserFromGroup

Removes the specified user from the specified group.

Calling this action 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. 450).

The request accepts the following data in JSON format.

GroupName (p. 59)

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. 59)

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. 59)

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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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.

> **Note**
> This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.
> If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In *sandbox mode* , you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see SMS message settings for Amazon Cognito user pools in the *Amazon Cognito Developer Guide*.

Calling this action requires developer credentials.

Request Syntax

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

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 450).

The request accepts the following data in JSON format.

ClientMetadata (p. 61)

A map of custom key-value pairs that you can provide as input for any custom workflows that this action triggers.

You create custom workflows by assigning AWS Lambda functions to user pool triggers. When you use the AdminResetUserPassword API action, Amazon Cognito invokes the function that is assigned to the *custom message* trigger. When Amazon Cognito invokes this function, it passes a JSON payload, which the function receives as input. This payload contains a `clientMetadata` attribute, which provides the data that you assigned to the ClientMetadata parameter in your AdminResetUserPassword request. In your function code in AWS Lambda, you can process the `clientMetadata` value to enhance your workflow for your specific needs.

For more information, see Customizing user pool Workflows with Lambda Triggers in the *Amazon Cognito Developer Guide*.

> **Note**
>
> When you use the ClientMetadata parameter, remember that Amazon Cognito won't do the following:
>
> - Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn't include triggers, the ClientMetadata parameter serves no purpose.
> - Validate the ClientMetadata value.
> - Encrypt the ClientMetadata value. Don't use Amazon Cognito to provide sensitive information.

Type: String to string map

Required: No

Username (p. 61)

The user name of the user whose password you want 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. 61)

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. 452).

InternalErrorException

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

HTTP Status Code: 500

InvalidEmailRoleAccessPolicyException

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

HTTP Status Code: 400

InvalidLambdaResponseException

This exception is thrown when Amazon Cognito 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 doesn't have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't 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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 Amazon Cognito encounters an unexpected exception with AWS Lambda.

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 isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

AdminRespondToAuthChallenge

Responds to an authentication challenge, as an administrator.

> **Note**
> This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.
> If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In *sandbox mode* , you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see SMS message settings for Amazon Cognito user pools in the *Amazon Cognito Developer Guide*.

Calling this action requires developer credentials.

Request Syntax

```
{
   "AnalyticsMetadata": {
      "AnalyticsEndpointId": "string"
   },
   "ChallengeName": "string",
   "ChallengeResponses": {
      "string" : "string"
   },
   "ClientId": "string",
   "ClientMetadata": {
      "string" : "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. 450).

The request accepts the following data in JSON format.

AnalyticsMetadata (p. 65)

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

Type: AnalyticsMetadataType (p. 358) object

Required: No

ChallengeName (p. 65)

The challenge name. For more information, see AdminInitiateAuth.

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. 65)

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).

  Note

  `PASSWORD_VERIFIER` requires `DEVICE_KEY` when signing in with a remembered device.
- `ADMIN_NO_SRP_AUTH`: `PASSWORD`, `USERNAME`, `SECRET_HASH` (if app client is configured with client secret).
- `NEW_PASSWORD_REQUIRED`: `NEW_PASSWORD`, `USERNAME`, `SECRET_HASH` (if app client is configured with client secret). To set any required attributes that Amazon Cognito returned as `requiredAttributes` in the `AdminInitiateAuth` response, add a `userAttributes.`*`attributename`* parameter. This parameter can also set values for writable attributes that aren't required by your user pool.

  Note

  In a `NEW_PASSWORD_REQUIRED` challenge response, you can't modify a required attribute that already has a value. In `AdminRespondToAuthChallenge`, set a value for any keys that Amazon Cognito returned in the `requiredAttributes` parameter, then use the `AdminUpdateUserAttributes` API operation to modify the value of any additional attributes.
- `MFA_SETUP` requires `USERNAME`, plus you must use the session value returned by `VerifySoftwareToken` in the `Session` parameter.

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

Type: String to string map

Required: No

ClientId (p. 65)

The app client ID.

Type: String

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

Pattern: `[\w+]+`

Required: Yes

ClientMetadata (p. 65)

A map of custom key-value pairs that you can provide as input for any custom workflows that this action triggers.

You create custom workflows by assigning AWS Lambda functions to user pool triggers. When you use the AdminRespondToAuthChallenge API action, Amazon Cognito invokes any functions that you have assigned to the following triggers:

- pre sign-up
- custom message
- post authentication
- user migration
- pre token generation
- define auth challenge
- create auth challenge
- verify auth challenge response

When Amazon Cognito invokes any of these functions, it passes a JSON payload, which the function receives as input. This payload contains a `clientMetadata` attribute that provides the data that you assigned to the ClientMetadata parameter in your AdminRespondToAuthChallenge request. In your function code in AWS Lambda, you can process the `clientMetadata` value to enhance your workflow for your specific needs.

For more information, see Customizing user pool Workflows with Lambda Triggers in the *Amazon Cognito Developer Guide*.

> **Note**
> When you use the ClientMetadata parameter, remember that Amazon Cognito won't do the following:
>
> - Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn't include triggers, the ClientMetadata parameter serves no purpose.
> - Validate the ClientMetadata value.
> - Encrypt the ClientMetadata value. Don't use Amazon Cognito to provide sensitive information.

Type: String to string map

Required: No

ContextData (p. 65)

Contextual data about your user session, such as the device fingerprint, IP address, or location. Amazon Cognito advanced security evaluates the risk of an authentication event based on the context that your app generates and passes to Amazon Cognito when it makes API requests.

Type: ContextDataType (p. 368) object

Required: No

Session (p. 65)

The session that should be passed both ways in challenge-response calls to the service. If an `InitiateAuth` or `RespondToAuthChallenge` API call determines that the caller must pass another challenge, it returns 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. 65)

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. 68)

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

Type: AuthenticationResultType (p. 360) object

ChallengeName (p. 68)

The name of the challenge. For more information, see AdminInitiateAuth.

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. 68)

The challenge parameters. For more information, see AdminInitiateAuth.

Type: String to string map

Session (p. 68)

The session that should be passed both ways in challenge-response calls to the service. If the caller must pass 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. 452).

AliasExistsException

This exception is thrown when a user tries to confirm the account with an email address or phone number that has already been supplied as an alias for a different user profile. This exception indicates that an account with this email address or phone already exists in a user pool that you've configured to use email address or phone number as a sign-in alias.

HTTP Status Code: 400

CodeMismatchException

This exception is thrown if the provided code doesn't 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 Amazon Cognito 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 Amazon Cognito encounters an invalid password.

HTTP Status Code: 400

InvalidSmsRoleAccessPolicyException

This exception is returned when the role provided for SMS configuration doesn't have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't 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 not valid.

HTTP Status Code: 400

MFAMethodNotFoundException

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

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't 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 can't find the requested resource.

HTTP Status Code: 400

SoftwareTokenMFANotFoundException

This exception is thrown when the software token time-based one-time password (TOTP) multi-factor authentication (MFA) isn't activated 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 Amazon Cognito encounters an unexpected exception with AWS Lambda.

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 isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

AdminSetUserMFAPreference

The user's multi-factor authentication (MFA) preference, including which MFA options are activated, and if any are preferred. Only one factor can be set as preferred. The preferred MFA factor will be used to authenticate a user if multiple factors are activated. If multiple options are activated and no preference is set, a challenge to choose an MFA option will be returned during sign-in.

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. 450).

The request accepts the following data in JSON format.

SMSMfaSettings (p. 72)

The SMS text message MFA settings.

Type: SMSMfaSettingsType (p. 415) object

Required: No

SoftwareTokenMfaSettings (p. 72)

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

Type: SoftwareTokenMfaSettingsType (p. 417) object

Required: No

Username (p. 72)

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. 72)

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. 452).

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 isn't 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 can't find the requested resource.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

AdminSetUserPassword

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

The password can be temporary or permanent. If it is temporary, the user status enters the `FORCE_CHANGE_PASSWORD` state. When the user next tries to sign in, the InitiateAuth/AdminInitiateAuth response will contain the `NEW_PASSWORD_REQUIRED` challenge. If the user doesn't sign in before it expires, the user won't be able to sign in, and an administrator must reset their password.

Once the user has set a new password, or the password is permanent, the user status is set to `Confirmed`.

Request Syntax

```
{
   "Password": "string",
   "Permanent": boolean,
   "Username": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 450).

The request accepts the following data in JSON format.

Password (p. 75)

The password for the user.

Type: String

Length Constraints: Maximum length of 256.

Pattern: `[\S]+`

Required: Yes

Permanent (p. 75)

`True` if the password is permanent, `False` if it is temporary.

Type: Boolean

Required: No

Username (p. 75)

The user name of the user whose password you want to set.

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. 75)

The user pool ID for the user pool where you want to set 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. 452).

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 Amazon Cognito encounters an invalid password.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

AdminSetUserSettings

This action is no longer supported. You can use it to configure only SMS MFA. You can't use it to configure time-based one-time password (TOTP) software token MFA. To configure either type of MFA, use AdminSetUserMFAPreference instead.

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. 450).

The request accepts the following data in JSON format.

MFAOptions (p. 78)

You can use this parameter only to set an SMS configuration that uses SMS for delivery.

Type: Array of MFAOptionType (p. 393) objects

Required: Yes

Username (p. 78)

The user name of the user whose options you're setting.

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. 78)

The ID of the user pool that contains the user whose options you're setting.

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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

AdminUpdateAuthEventFeedback

Provides feedback for an authentication event indicating if 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. 450).

The request accepts the following data in JSON format.

EventId (p. 80)

The authentication event ID.

Type: String

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

Pattern: `[\w+-]+`

Required: Yes

FeedbackValue (p. 80)

The authentication event feedback value.

Type: String

Valid Values: `Valid | Invalid`

Required: Yes

Username (p. 80)

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. 80)

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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 isn't found.

HTTP Status Code: 400

UserPoolAddOnNotEnabledException

This exception is thrown when user pool add-ons aren't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

AdminUpdateDeviceStatus

Updates the device status as an administrator.

Calling this action 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. 450).

The request accepts the following data in JSON format.

DeviceKey (p. 83)

The device key.

Type: String

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

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

Required: Yes

DeviceRememberedStatus (p. 83)

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

Type: String

Valid Values: `remembered | not_remembered`

Required: No

Username (p. 83)

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. 83)

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. 452).

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 not valid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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.

> **Note**
> This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.
> If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In *sandbox mode* , you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see SMS message settings for Amazon Cognito user pools in the *Amazon Cognito Developer Guide*.

Calling this action requires developer credentials.

Request Syntax

```
{
   "ClientMetadata": {
      "string" : "string"
   },
   "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. 450).

The request accepts the following data in JSON format.

ClientMetadata (p. 86)

A map of custom key-value pairs that you can provide as input for any custom workflows that this action triggers.

You create custom workflows by assigning AWS Lambda functions to user pool triggers. When you use the AdminUpdateUserAttributes API action, Amazon Cognito invokes the function that is assigned to the *custom message* trigger. When Amazon Cognito invokes this function, it passes a JSON payload, which the function receives as input. This payload contains a `clientMetadata` attribute, which provides the data that you assigned to the ClientMetadata parameter in your AdminUpdateUserAttributes request. In your function code in AWS Lambda, you can process the `clientMetadata` value to enhance your workflow for your specific needs.

For more information, see Customizing user pool Workflows with Lambda Triggers in the *Amazon Cognito Developer Guide*.

> **Note**
> When you use the ClientMetadata parameter, remember that Amazon Cognito won't do the following:
>
> - Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn't include triggers, the ClientMetadata parameter serves no purpose.
> - Validate the ClientMetadata value.
> - Encrypt the ClientMetadata value. Don't use Amazon Cognito to provide sensitive information.

Type: String to string map

Required: No

UserAttributes (p. 86)

An array of name-value pairs representing user attributes.

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

If your user pool requires verification before Amazon Cognito updates an attribute value that you specify in this request, Amazon Cognito doesn’t immediately update the value of that attribute. After your user receives and responds to a verification message to verify the new value, Amazon Cognito updates the attribute value. Your user can sign in and receive messages with the original attribute value until they verify the new value.

To update the value of an attribute that requires verification in the same API request, include the `email_verified` or `phone_number_verified` attribute, with a value of `true`. If you set the `email_verified` or `phone_number_verified` value for an `email` or `phone_number` attribute that requires verification to `true`, Amazon Cognito doesn’t send a verification message to your user.

Type: Array of AttributeType (p. 359) objects

Required: Yes

Username (p. 86)

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. 86)

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. 452).

AliasExistsException

This exception is thrown when a user tries to confirm the account with an email address or phone number that has already been supplied as an alias for a different user profile. This exception indicates that an account with this email address or phone already exists in a user pool that you've configured to use email address or phone number as a sign-in alias.

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 isn't allowed to use your email identity. HTTP status code: 400.

HTTP Status Code: 400

InvalidLambdaResponseException

This exception is thrown when Amazon Cognito 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 doesn't have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't 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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 Amazon Cognito encounters an unexpected exception with AWS Lambda.

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 isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

AdminUserGlobalSignOut

Signs out a user from all devices. You must sign `AdminUserGlobalSignOut` requests with AWS credentials. It also invalidates all refresh tokens that Amazon Cognito has issued to a user. The user's current access and ID tokens remain valid until they expire. By default, access and ID tokens expire one hour after they're issued. A user can still use a hosted UI cookie to retrieve new tokens for the duration of the cookie validity period of 1 hour.

Calling this action 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. 450).

The request accepts the following data in JSON format.

Username (p. 90)

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. 90)

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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

AssociateSoftwareToken

Begins setup of time-based one-time password multi-factor authentication (TOTP MFA) for a user, with a unique private key that Amazon Cognito generates and returns in the API response. You can authorize an `AssociateSoftwareToken` request with either the user's access token, or a session string from a challenge response that you received from Amazon Cognito.

> **Note**
> Amazon Cognito disassociates an existing software token when you verify the new token in a VerifySoftwareToken API request. If you don't verify the software token and your user pool doesn't require MFA, the user can then authenticate with user name and password credentials alone. If your user pool requires TOTP MFA, Amazon Cognito generates an `MFA_SETUP` or `SOFTWARE_TOKEN_SETUP` challenge each time your user signs. Complete setup with `AssociateSoftwareToken` and `VerifySoftwareToken`.
> After you set up software token MFA for your user, Amazon Cognito generates a `SOFTWARE_TOKEN_MFA` challenge when they authenticate. Respond to this challenge with your user's TOTP.

Request Syntax

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

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 450).

The request accepts the following data in JSON format.

AccessToken (p. 92)

A valid access token that Amazon Cognito issued to the user whose software token you want to generate.

Type: String

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

Required: No

Session (p. 92)

The session that 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. 92)

A unique generated shared secret code that is used in the time-based one-time password (TOTP) algorithm to generate a one-time code.

Type: String

Length Constraints: Minimum length of 16.

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

Session (p. 92)

The session that 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. 452).

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

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

SoftwareTokenMFANotFoundException

This exception is thrown when the software token time-based one-time password (TOTP) multi-factor authentication (MFA) isn't activated 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

AccessToken (p. 95)

A valid access token that Amazon Cognito issued to the user whose password you want to change.

Type: String

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

Required: Yes

PreviousPassword (p. 95)

The old password.

Type: String

Length Constraints: Maximum length of 256.

Pattern: `[\S]+`

Required: Yes

ProposedPassword (p. 95)

The new password.

Type: String

Length Constraints: 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. 452).

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 Amazon Cognito 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 isn't 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 can't 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 isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

AccessToken (p. 98)

A valid access token that Amazon Cognito issued to the user whose device you want to confirm.

Type: String

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

Required: Yes

DeviceKey (p. 98)

The device key.

Type: String

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

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

Required: Yes

DeviceName (p. 98)

The device name.

Type: String

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

Required: No

DeviceSecretVerifierConfig (p. 98)

The configuration of the device secret verifier.

Type: DeviceSecretVerifierConfigType (p. 373) 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. 99)

Indicates whether the user confirmation must confirm the device response.

Type: Boolean

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

InternalErrorException

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

HTTP Status Code: 500

InvalidLambdaResponseException

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

HTTP Status Code: 400

InvalidUserPoolConfigurationException

This exception is thrown when the user pool configuration is not valid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't 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 can't 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 isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

ConfirmForgotPassword

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

Request Syntax

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

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 450).

The request accepts the following data in JSON format.

AnalyticsMetadata (p. 101)

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

Type: AnalyticsMetadataType (p. 358) object

Required: No

ClientId (p. 101)

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

ClientMetadata (p. 101)

A map of custom key-value pairs that you can provide as input for any custom workflows that this action triggers.

You create custom workflows by assigning AWS Lambda functions to user pool triggers. When you use the ConfirmForgotPassword API action, Amazon Cognito invokes the function that is assigned to the *post confirmation* trigger. When Amazon Cognito invokes this function, it passes a JSON payload, which the function receives as input. This payload contains a `clientMetadata` attribute, which provides the data that you assigned to the ClientMetadata parameter in your ConfirmForgotPassword request. In your function code in AWS Lambda, you can process the `clientMetadata` value to enhance your workflow for your specific needs.

For more information, see Customizing user pool Workflows with Lambda Triggers in the *Amazon Cognito Developer Guide*.

> **Note**
> When you use the ClientMetadata parameter, remember that Amazon Cognito won't do the following:
>
> - Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn't include triggers, the ClientMetadata parameter serves no purpose.
> - Validate the ClientMetadata value.
> - Encrypt the ClientMetadata value. Don't use Amazon Cognito to provide sensitive information.

Type: String to string map

Required: No

ConfirmationCode (p. 101)

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

Type: String

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

Pattern: `[\S]+`

Required: Yes

Password (p. 101)

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

Type: String

Length Constraints: Maximum length of 256.

Pattern: `[\S]+`

Required: Yes

SecretHash (p. 101)

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. 101)

Contextual data about your user session, such as the device fingerprint, IP address, or location. Amazon Cognito advanced security evaluates the risk of an authentication event based on the context that your app generates and passes to Amazon Cognito when it makes API requests.

Type: UserContextDataType (p. 423) object

Required: No

Username (p. 101)

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. 452).

CodeMismatchException

This exception is thrown if the provided code doesn't 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 Amazon Cognito 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 Amazon Cognito 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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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, such as 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 Amazon Cognito encounters an unexpected exception with AWS Lambda.

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 isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't 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 V2
- AWS SDK for JavaScript

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

ConfirmSignUp

Confirms registration of a new user.

Request Syntax

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

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 450).

The request accepts the following data in JSON format.

AnalyticsMetadata (p. 106)

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

Type: AnalyticsMetadataType (p. 358) object

Required: No

ClientId (p. 106)

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

ClientMetadata (p. 106)

A map of custom key-value pairs that you can provide as input for any custom workflows that this action triggers.

You create custom workflows by assigning AWS Lambda functions to user pool triggers. When you use the ConfirmSignUp API action, Amazon Cognito invokes the function that is assigned to the *post confirmation* trigger. When Amazon Cognito invokes this function, it passes a JSON payload, which the function receives as input. This payload contains a `clientMetadata` attribute, which provides the data that you assigned to the ClientMetadata parameter in your ConfirmSignUp request. In your function code in AWS Lambda, you can process the `clientMetadata` value to enhance your workflow for your specific needs.

For more information, see Customizing user pool Workflows with Lambda Triggers in the *Amazon Cognito Developer Guide*.

> **Note**
> When you use the ClientMetadata parameter, remember that Amazon Cognito won't do the following:
>
> - Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn't include triggers, the ClientMetadata parameter serves no purpose.
> - Validate the ClientMetadata value.
> - Encrypt the ClientMetadata value. Don't use Amazon Cognito to provide sensitive information.

Type: String to string map

Required: No

ConfirmationCode (p. 106)

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. 106)

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. 106)

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. 106)

Contextual data about your user session, such as the device fingerprint, IP address, or location. Amazon Cognito advanced security evaluates the risk of an authentication event based on the context that your app generates and passes to Amazon Cognito when it makes API requests.

Type: UserContextDataType (p. 423) object

Required: No

Username (p. 106)

The user name of the user whose registration you want 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. 452).

AliasExistsException

This exception is thrown when a user tries to confirm the account with an email address or phone number that has already been supplied as an alias for a different user profile. This exception indicates that an account with this email address or phone already exists in a user pool that you've configured to use email address or phone number as a sign-in alias.

HTTP Status Code: 400

CodeMismatchException

This exception is thrown if the provided code doesn't 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 Amazon Cognito 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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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, such as 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 Amazon Cognito encounters an unexpected exception with AWS Lambda.

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 isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3

- AWS SDK for Python
- AWS SDK for Ruby V3

CreateGroup

Creates a new group in the specified user pool.

Calling this action 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. 450).

The request accepts the following data in JSON format.

Description (p. 111)

A string containing the description of the group.

Type: String

Length Constraints: Maximum length of 2048.

Required: No

GroupName (p. 111)

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. 111)

A non-negative 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 is given in the user's tokens for the `cognito:roles` and `cognito:preferred_role` claims.

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 isn't set in users' tokens.

The default `Precedence` value is null. The maximum `Precedence` value is `2^31-1`.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

RoleArn (p. 111)

The role Amazon Resource Name (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. 111)

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. 112)

The group object for the group.

Type: GroupType (p. 383) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CreateIdentityProvider

Creates an IdP 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. 450).

The request accepts the following data in JSON format.

AttributeMapping (p. 114)

A mapping of IdP 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. 114)

A list of IdP 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. 114)

The IdP details. The following list describes the provider detail keys for each IdP type.

- For Google and Login with Amazon:
  - client_id
  - client_secret
  - authorize_scopes
- For Facebook:

    - client_id
    - client_secret
    - authorize_scopes
    - api_version
- For Sign in with Apple:
    - client_id
    - team_id
    - key_id
    - private_key
    - authorize_scopes
- For OpenID Connect (OIDC) providers:
    - client_id
    - client_secret
    - attributes_request_method
    - oidc_issuer
    - authorize_scopes
    - The following keys are only present if Amazon Cognito didn't discover them at the `oidc_issuer` URL.
        - authorize_url
        - token_url
        - attributes_url
        - jwks_uri
    - Amazon Cognito sets the value of the following keys automatically. They are read-only.
        - attributes_url_add_attributes
- For SAML providers:
    - MetadataFile or MetadataURL
    - IDPSignout *optional*

Type: String to string map

Required: Yes

ProviderName (p. 114)

The IdP name.

Type: String

Length Constraints: Minimum length of 3. Maximum length of 32.

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

Required: Yes

ProviderType (p. 114)

The IdP type.

Type: String

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

Required: Yes

UserPoolId (p. 114)

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. 116)

The newly created IdP object.

Type: IdentityProviderType (p. 386) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CreateResourceServer

Creates a new OAuth2.0 resource server and defines custom scopes within 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. 450).

The request accepts the following data in JSON format.

Identifier (p. 118)

A unique resource server identifier for the resource server. This could be an HTTPS endpoint where the resource server is located, such as `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. 118)

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. 118)

A list of scopes. Each scope is a key-value map with the keys `name` and `description`.

Type: Array of ResourceServerScopeType (p. 404) objects

Array Members: Maximum number of 100 items.

Required: No

UserPoolId (p. 118)

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. 119)

The newly created resource server.

Type: ResourceServerType (p. 405) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

CloudWatchLogsRoleArn (p. 121)

The role ARN for the Amazon CloudWatch Logs 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. 121)

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. 121)

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. 121)

The job object that represents the user import job.

Type: UserImportJobType (p. 424) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't 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 can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CreateUserPool

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

Note

This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.

If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In *sandbox mode* , you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see SMS message settings for Amazon Cognito user pools in the *Amazon Cognito Developer Guide*.

Request Syntax

```
{
   "AccountRecoverySetting": {
      "RecoveryMechanisms": [
         {
            "Name": "string",
            "Priority": number
         }
      ]
   },
   "AdminCreateUserConfig": {
      "AllowAdminCreateUserOnly": boolean,
      "InviteMessageTemplate": {
         "EmailMessage": "string",
         "EmailSubject": "string",
         "SMSMessage": "string"
      },
      "UnusedAccountValidityDays": number
   },
   "AliasAttributes": [ "string" ],
   "AutoVerifiedAttributes": [ "string" ],
   "DeviceConfiguration": {
      "ChallengeRequiredOnNewDevice": boolean,
      "DeviceOnlyRememberedOnUserPrompt": boolean
   },
   "EmailConfiguration": {
      "ConfigurationSet": "string",
      "EmailSendingAccount": "string",
      "From": "string",
      "ReplyToEmailAddress": "string",
      "SourceArn": "string"
   },
   "EmailVerificationMessage": "string",
   "EmailVerificationSubject": "string",
   "LambdaConfig": {
      "CreateAuthChallenge": "string",
      "CustomEmailSender": {
         "LambdaArn": "string",
         "LambdaVersion": "string"
      },
      "CustomMessage": "string",
      "CustomSMSSender": {
         "LambdaArn": "string",
         "LambdaVersion": "string"
      },
      "DefineAuthChallenge": "string",
      "KMSKeyID": "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,
         "TemporaryPasswordValidityDays": number
      }
   },
   "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",
      "SnsRegion": "string"
   },
   "SmsVerificationMessage": "string",
   "UserAttributeUpdateSettings": {
      "AttributesRequireVerificationBeforeUpdate": [ "string" ]
   },
   "UsernameAttributes": [ "string" ],
   "UsernameConfiguration": {
      "CaseSensitive": boolean
   },
   "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. 450).

The request accepts the following data in JSON format.

AccountRecoverySetting (p. 124)

The available verified method a user can use to recover their password when they call `ForgotPassword`. You can use this setting to define a preferred method when a user has more than one method available. With this setting, SMS doesn't qualify for a valid password recovery mechanism if the user also has SMS multi-factor authentication (MFA) activated. In the absence of this setting, Amazon Cognito uses the legacy behavior to determine the recovery method where SMS is preferred through email.

Type: AccountRecoverySettingType (p. 351) object

Required: No

AdminCreateUserConfig (p. 124)

The configuration for `AdminCreateUser` requests.

Type: AdminCreateUserConfigType (p. 355) object

Required: No

AliasAttributes (p. 124)

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. 124)

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

Type: Array of strings

Valid Values: `phone_number | email`

Required: No

DeviceConfiguration (p. 124)

The device configuration.

Type: DeviceConfigurationType (p. 372) object

Required: No

EmailConfiguration (p. 124)

The email configuration of your user pool. The email configuration type sets your preferred sending method, AWS Region, and sender for messages from your user pool.

Type: EmailConfigurationType (p. 377) object

Required: No

EmailVerificationMessage (p. 124)

A string representing the email verification message. EmailVerificationMessage is allowed only if EmailSendingAccount is DEVELOPER.

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. 124)

A string representing the email verification subject. EmailVerificationSubject is allowed only if EmailSendingAccount is DEVELOPER.

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. 124)

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 must 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. 389) object

Required: No

MfaConfiguration (p. 124)

Specifies MFA configuration details.

Type: String

Valid Values: `OFF | ON | OPTIONAL`

Required: No

Policies (p. 124)

The policies associated with the new user pool.

Type: UserPoolPolicyType (p. 439) object

Required: No

PoolName (p. 124)

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. 124)

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

Type: Array of SchemaAttributeType (p. 410) objects

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

Required: No

SmsAuthenticationMessage (p. 124)

A string representing the SMS authentication message.

Type: String

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

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

Required: No

SmsConfiguration (p. 124)

The SMS configuration with the settings that your Amazon Cognito user pool must use to send an SMS message from your AWS account through Amazon Simple Notification Service. To send SMS messages with Amazon SNS in the AWS Region that you want, the Amazon Cognito user pool uses an AWS Identity and Access Management (IAM) role in your AWS account.

Type: SmsConfigurationType (p. 412) object

Required: No

SmsVerificationMessage (p. 124)

A string representing the SMS verification message.

Type: String

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

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

Required: No

UserAttributeUpdateSettings (p. 124)

The settings for updates to user attributes. These settings include the property `AttributesRequireVerificationBeforeUpdate`, a user-pool setting that tells Amazon Cognito how to handle changes to the value of your users' email address and phone number attributes. For more information, see Verifying updates to to email addresses and phone numbers.

Type: UserAttributeUpdateSettingsType (p. 422) object

Required: No

UsernameAttributes (p. 124)

Specifies whether a user can use an email address or phone number as a username when they sign up.

Type: Array of strings

Valid Values: `phone_number | email`

Required: No

UsernameConfiguration (p. 124)

Case sensitivity on the username input for the selected sign-in option. For example, when case sensitivity is set to `False`, users can sign in using either "username" or "Username". This configuration is immutable once it has been set. For more information, see UsernameConfigurationType.

Type: UsernameConfigurationType (p. 427) object

Required: No

UserPoolAddOns (p. 124)

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

Type: UserPoolAddOnsType (p. 428) object

Required: No

UserPoolTags (p. 124)

The tag keys and values to assign to the user pool. A tag is a label that you can use to categorize and manage user pools in different ways, such as by purpose, owner, environment, or other criteria.

Type: String to string map

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

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

Required: No

VerificationMessageTemplate (p. 124)

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

Type: VerificationMessageTemplateType (p. 448) object

Required: No

Response Syntax

```
{
   "UserPool": {
      "AccountRecoverySetting": {
         "RecoveryMechanisms": [
            {
               "Name": "string",
               "Priority": number
            }
         ]
      },
      "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": {
         "ConfigurationSet": "string",
         "EmailSendingAccount": "string",
         "From": "string",
         "ReplyToEmailAddress": "string",
         "SourceArn": "string"
      },
      "EmailConfigurationFailure": "string",
      "EmailVerificationMessage": "string",
      "EmailVerificationSubject": "string",
      "EstimatedNumberOfUsers": number,
      "Id": "string",
      "LambdaConfig": {
         "CreateAuthChallenge": "string",
         "CustomEmailSender": {
            "LambdaArn": "string",
            "LambdaVersion": "string"
         },
         "CustomMessage": "string",
         "CustomSMSSender": {
            "LambdaArn": "string",
            "LambdaVersion": "string"
         },
         "DefineAuthChallenge": "string",
         "KMSKeyID": "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,
            "TemporaryPasswordValidityDays": number
         }
      },
      "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",
         "SnsRegion": "string"
      },
      "SmsConfigurationFailure": "string",
      "SmsVerificationMessage": "string",
      "Status": "string",
      "UserAttributeUpdateSettings": {
         "AttributesRequireVerificationBeforeUpdate": [ "string" ]
      },
      "UsernameAttributes": [ "string" ],
      "UsernameConfiguration": {
         "CaseSensitive": boolean
      },
      "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. 129)

A container for the user pool details.

Type: UserPoolType (p. 440) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

InternalErrorException

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

HTTP Status Code: 500

InvalidEmailRoleAccessPolicyException

This exception is thrown when Amazon Cognito isn't 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 doesn't have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't 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 isn't 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 can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

CreateUserPoolClient

Creates the user pool client.

When you create a new user pool client, token revocation is automatically activated. For more information about revoking tokens, see RevokeToken.

Request Syntax

```
{
   "AccessTokenValidity": number,
   "AllowedOAuthFlows": [ "string" ],
   "AllowedOAuthFlowsUserPoolClient": boolean,
   "AllowedOAuthScopes": [ "string" ],
   "AnalyticsConfiguration": {
      "ApplicationArn": "string",
      "ApplicationId": "string",
      "ExternalId": "string",
      "RoleArn": "string",
      "UserDataShared": boolean
   },
   "CallbackURLs": [ "string" ],
   "ClientName": "string",
   "DefaultRedirectURI": "string",
   "EnablePropagateAdditionalUserContextData": boolean,
   "EnableTokenRevocation": boolean,
   "ExplicitAuthFlows": [ "string" ],
   "GenerateSecret": boolean,
   "IdTokenValidity": number,
   "LogoutURLs": [ "string" ],
   "PreventUserExistenceErrors": "string",
   "ReadAttributes": [ "string" ],
   "RefreshTokenValidity": number,
   "SupportedIdentityProviders": [ "string" ],
   "TokenValidityUnits": {
      "AccessToken": "string",
      "IdToken": "string",
      "RefreshToken": "string"
   },
   "UserPoolId": "string",
   "WriteAttributes": [ "string" ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 450).

The request accepts the following data in JSON format.

AccessTokenValidity (p. 134)

The access token time limit. After this limit expires, your user can't use their access token. To specify the time unit for `AccessTokenValidity` as `seconds`, `minutes`, `hours`, or `days`, set a `TokenValidityUnits` value in your API request.

For example, when you set `AccessTokenValidity` to `10` and `TokenValidityUnits` to `hours`, your user can authorize access with their access token for 10 hours.

The default time unit for `AccessTokenValidity` in an API request is hours. *Valid range* is displayed below in seconds.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 86400.

Required: No

AllowedOAuthFlows (p. 134)

The allowed OAuth flows.

code

Use a code grant flow, which provides an authorization code as the response. This code can be exchanged for access tokens with the `/oauth2/token` endpoint.

implicit

Issue the access token (and, optionally, ID token, based on scopes) directly to your user.

client_credentials

Issue the access token from the `/oauth2/token` endpoint directly to a non-person user using a combination of the client ID and client secret.

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. 134)

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

Type: Boolean

Required: No

AllowedOAuthScopes (p. 134)

The allowed OAuth scopes. Possible values provided by OAuth are `phone`, `email`, `openid`, and `profile`. Possible values provided by AWS are `aws.cognito.signin.user.admin`. Custom scopes created in Resource Servers are also supported.

Type: Array of strings

Array Members: Maximum number of 50 items.

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

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

Required: No

AnalyticsConfiguration (p. 134)

The user pool analytics configuration for collecting metrics and sending them to your Amazon Pinpoint campaign.

Note

In AWS Regions where Amazon Pinpoint isn't available, user pools only support sending events to Amazon Pinpoint projects in AWS Region us-east-1. In Regions where Amazon

Pinpoint is available, user pools support sending events to Amazon Pinpoint projects within that same Region.

Type: AnalyticsConfigurationType (p. 356) object

Required: No

CallbackURLs (p. 134)

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

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. 134)

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. 134)

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

EnablePropagateAdditionalUserContextData (p. 134)

Activates the propagation of additional user context data. For more information about propagation of user context data, see Adding advanced security to a user pool. If you don’t include this parameter, you can't send device fingerprint information, including source IP address, to Amazon Cognito advanced security. You can only activate `EnablePropagateAdditionalUserContextData` in an app client that has a client secret.

Type: Boolean

Required: No

EnableTokenRevocation (p. 134)

Activates or deactivates token revocation. For more information about revoking tokens, see RevokeToken.

If you don't include this parameter, token revocation is automatically activated for the new user pool client.

Type: Boolean

Required: No

ExplicitAuthFlows (p. 134)

The authentication flows that are supported by the user pool clients. Flow names without the `ALLOW_` prefix are no longer supported, in favor of new names with the `ALLOW_` prefix.

Note
Values with `ALLOW_` prefix must be used only along with the `ALLOW_` prefix.

Valid values include:

- `ALLOW_ADMIN_USER_PASSWORD_AUTH`: Enable admin based user password authentication flow `ADMIN_USER_PASSWORD_AUTH`. This setting replaces the `ADMIN_NO_SRP_AUTH` setting. With this authentication flow, Amazon Cognito receives the password in the request instead of using the Secure Remote Password (SRP) protocol to verify passwords.
- `ALLOW_CUSTOM_AUTH`: Enable AWS Lambda trigger based authentication.
- `ALLOW_USER_PASSWORD_AUTH`: Enable user password-based authentication. In this flow, Amazon Cognito receives the password in the request instead of using the SRP protocol to verify passwords.
- `ALLOW_USER_SRP_AUTH`: Enable SRP-based authentication.
- `ALLOW_REFRESH_TOKEN_AUTH`: Enable authflow to refresh tokens.

If you don't specify a value for `ExplicitAuthFlows`, your app client activates the `ALLOW_USER_SRP_AUTH` and `ALLOW_CUSTOM_AUTH` authentication flows.

Type: Array of strings

Valid Values: `ADMIN_NO_SRP_AUTH | CUSTOM_AUTH_FLOW_ONLY | USER_PASSWORD_AUTH | ALLOW_ADMIN_USER_PASSWORD_AUTH | ALLOW_CUSTOM_AUTH | ALLOW_USER_PASSWORD_AUTH | ALLOW_USER_SRP_AUTH | ALLOW_REFRESH_TOKEN_AUTH`

Required: No

GenerateSecret (p. 134)

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

Type: Boolean

Required: No

IdTokenValidity (p. 134)

The ID token time limit. After this limit expires, your user can't use their ID token. To specify the time unit for `IdTokenValidity` as `seconds`, `minutes`, `hours`, or `days`, set a `TokenValidityUnits` value in your API request.

For example, when you set `IdTokenValidity` as `10` and `TokenValidityUnits` as `hours`, your user can authenticate their session with their ID token for 10 hours.

The default time unit for `AccessTokenValidity` in an API request is hours. *Valid range* is displayed below in seconds.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 86400.

Required: No

LogoutURLs (p. 134)

A list of allowed logout URLs for the IdPs.

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

PreventUserExistenceErrors (p. 134)

Errors and responses that you want Amazon Cognito APIs to return during authentication, account confirmation, and password recovery when the user doesn't exist in the user pool. When set to `ENABLED` and the user doesn't exist, authentication returns an error indicating either the username or password was incorrect. Account confirmation and password recovery return a response indicating a code was sent to a simulated destination. When set to `LEGACY`, those APIs return a `UserNotFoundException` exception if the user doesn't exist in the user pool.

Valid values include:

- `ENABLED` - This prevents user existence-related errors.
- `LEGACY` - This represents the early behavior of Amazon Cognito where user existence related errors aren't prevented.

This setting affects the behavior of following APIs:

- AdminInitiateAuth (p. 39)
- AdminRespondToAuthChallenge (p. 65)
- InitiateAuth (p. 219)
- RespondToAuthChallenge (p. 261)
- ForgotPassword (p. 186)
- ConfirmForgotPassword (p. 101)
- ConfirmSignUp (p. 106)
- ResendConfirmationCode (p. 256)

Type: String

Valid Values: `LEGACY | ENABLED`

Required: No

ReadAttributes (p. 134)

The read attributes.

Type: Array of strings

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

Required: No

RefreshTokenValidity (p. 134)

The refresh token time limit. After this limit expires, your user can't use their refresh token. To specify the time unit for `RefreshTokenValidity` as `seconds`, `minutes`, `hours`, or `days`, set a `TokenValidityUnits` value in your API request.

For example, when you set `RefreshTokenValidity` as `10` and `TokenValidityUnits` as `days`, your user can refresh their session and retrieve new access and ID tokens for 10 days.

The default time unit for `RefreshTokenValidity` in an API request is days. You can't set `RefreshTokenValidity` to 0. If you do, Amazon Cognito overrides the value with the default value of 30 days. *Valid range* is displayed below in seconds.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 315360000.

Required: No

SupportedIdentityProviders (p. 134)

A list of provider names for the IdPs that this client supports. The following are supported: `COGNITO`, `Facebook`, `Google` `LoginWithAmazon`, and the names of your own SAML and OIDC providers.

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

TokenValidityUnits (p. 134)

The units in which the validity times are represented. The default unit for RefreshToken is days, and default for ID and access tokens are hours.

Type: TokenValidityUnitsType (p. 419) object

Required: No

UserPoolId (p. 134)

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. 134)

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

If your app client allows users to sign in through an IdP, this array must include all attributes that you have mapped to IdP attributes. Amazon Cognito updates mapped attributes when users sign in to your application through an IdP. If your app client does not have write access to a mapped attribute, Amazon Cognito throws an error when it tries to update the attribute. For more information, see Specifying IdP 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": {
      "AccessTokenValidity": number,
      "AllowedOAuthFlows": [ "string" ],
      "AllowedOAuthFlowsUserPoolClient": boolean,
      "AllowedOAuthScopes": [ "string" ],
      "AnalyticsConfiguration": {
         "ApplicationArn": "string",
         "ApplicationId": "string",
         "ExternalId": "string",
         "RoleArn": "string",
         "UserDataShared": boolean
      },
      "CallbackURLs": [ "string" ],
      "ClientId": "string",
      "ClientName": "string",
      "ClientSecret": "string",
      "CreationDate": number,
      "DefaultRedirectURI": "string",
      "EnablePropagateAdditionalUserContextData": boolean,
      "EnableTokenRevocation": boolean,
      "ExplicitAuthFlows": [ "string" ],
      "IdTokenValidity": number,
      "LastModifiedDate": number,
      "LogoutURLs": [ "string" ],
      "PreventUserExistenceErrors": "string",
      "ReadAttributes": [ "string" ],
      "RefreshTokenValidity": number,
      "SupportedIdentityProviders": [ "string" ],
      "TokenValidityUnits": {
         "AccessToken": "string",
         "IdToken": "string",
         "RefreshToken": "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. 140)

The user pool client that was just created.

Type: UserPoolClientType (p. 430) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 not valid.

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

ScopeDoesNotExistException

This exception is thrown when the specified scope doesn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

CustomDomainConfig (p. 143)

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. 369) object

Required: No

Domain (p. 143)

The domain string. For custom domains, this is the fully-qualified domain name, such as `auth.example.com`. For Amazon Cognito prefix domains, this is the prefix alone, such as `auth`.

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. 143)

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. 144)

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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

DeleteGroup

Deletes a group.

Calling this action 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. 450).

The request accepts the following data in JSON format.

GroupName (p. 146)

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. 146)

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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

DeleteIdentityProvider

Deletes an IdP 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. 450).

The request accepts the following data in JSON format.

ProviderName (p. 148)

The IdP 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. 148)

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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

Identifier (p. 150)

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. 150)

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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

AccessToken (p. 152)

A valid access token that Amazon Cognito issued to the user whose user profile you want to delete.

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. 452).

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 isn't 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 can't 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 isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

AccessToken (p. 154)

A valid access token that Amazon Cognito issued to the user whose attributes you want to delete.

Type: String

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

Required: Yes

UserAttributeNames (p. 154)

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

For custom attributes, you must prependattach the `custom:` prefix to the front of 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. 452).

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 isn't 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 can't 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 isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

UserPoolId (p. 156)

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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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're 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

ClientId (p. 158)

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. 158)

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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

Domain (p. 160)

The domain string. For custom domains, this is the fully-qualified domain name, such as `auth.example.com`. For Amazon Cognito prefix domains, this is the prefix alone, such as `auth`.

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. 160)

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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

DescribeIdentityProvider

Gets information about a specific IdP.

Request Syntax

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

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 450).

The request accepts the following data in JSON format.

ProviderName (p. 162)

The IdP 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. 162)

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. 162)

The IdP that was deleted.

Type: IdentityProviderType (p. 386) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

Identifier (p. 165)

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. 165)

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. 165)

The resource server.

Type: ResourceServerType (p. 405) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python

- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

ClientId (p. 168)

The app client ID.

Type: String

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

Pattern: `[\w+]+`

Required: No

UserPoolId (p. 168)

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. 168)

The risk configuration.

Type: RiskConfigurationType (p. 407) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 aren't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

JobId (p. 171)

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. 171)

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. 171)

The job object that represents the user import job.

Type: UserImportJobType (p. 424) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

UserPoolId (p. 174)

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": {
      "AccountRecoverySetting": {
         "RecoveryMechanisms": [
            {
               "Name": "string",
               "Priority": number
            }
         ]
      },
      "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": {
      "ConfigurationSet": "string",
      "EmailSendingAccount": "string",
      "From": "string",
      "ReplyToEmailAddress": "string",
      "SourceArn": "string"
   },
   "EmailConfigurationFailure": "string",
   "EmailVerificationMessage": "string",
   "EmailVerificationSubject": "string",
   "EstimatedNumberOfUsers": number,
   "Id": "string",
   "LambdaConfig": {
      "CreateAuthChallenge": "string",
      "CustomEmailSender": {
         "LambdaArn": "string",
         "LambdaVersion": "string"
      },
      "CustomMessage": "string",
      "CustomSMSSender": {
         "LambdaArn": "string",
         "LambdaVersion": "string"
      },
      "DefineAuthChallenge": "string",
      "KMSKeyID": "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,
         "TemporaryPasswordValidityDays": number
      }
   },
   "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",
         "SnsRegion": "string"
      },
      "SmsConfigurationFailure": "string",
      "SmsVerificationMessage": "string",
      "Status": "string",
      "UserAttributeUpdateSettings": {
         "AttributesRequireVerificationBeforeUpdate": [ "string" ]
      },
      "UsernameAttributes": [ "string" ],
      "UsernameConfiguration": {
         "CaseSensitive": boolean
      },
      "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. 174)

The container of metadata returned by the server to describe the pool.

Type: UserPoolType (p. 440) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

ClientId (p. 178)

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. 178)

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": {
      "AccessTokenValidity": number,
      "AllowedOAuthFlows": [ "string" ],
      "AllowedOAuthFlowsUserPoolClient": boolean,
      "AllowedOAuthScopes": [ "string" ],
      "AnalyticsConfiguration": {
         "ApplicationArn": "string",
         "ApplicationId": "string",
         "ExternalId": "string",
         "RoleArn": "string",
         "UserDataShared": boolean
      },
      "CallbackURLs": [ "string" ],
      "ClientId": "string",
      "ClientName": "string",
      "ClientSecret": "string",
      "CreationDate": number,
      "DefaultRedirectURI": "string",
      "EnablePropagateAdditionalUserContextData": boolean,
      "EnableTokenRevocation": boolean,
      "ExplicitAuthFlows": [ "string" ],
      "IdTokenValidity": number,
      "LastModifiedDate": number,
      "LogoutURLs": [ "string" ],
      "PreventUserExistenceErrors": "string",
      "ReadAttributes": [ "string" ],
      "RefreshTokenValidity": number,
      "SupportedIdentityProviders": [ "string" ],
      "TokenValidityUnits": {
         "AccessToken": "string",
         "IdToken": "string",
         "RefreshToken": "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. 178)

The user pool client from a server response to describe the user pool client.

Type: UserPoolClientType (p. 430) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

Domain (p. 181)

The domain string. For custom domains, this is the fully-qualified domain name, such as `auth.example.com`. For Amazon Cognito prefix domains, this is the prefix alone, such as `auth`.

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. 181)

A domain description object containing information about the domain.

Type: DomainDescriptionType (p. 375) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

AccessToken (p. 183)

A valid access token that Amazon Cognito issued to the user whose registered device you want to forget.

Type: String

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

Required: No

DeviceKey (p. 183)

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. 452).

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 not valid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't 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 can't 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 isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. The method used to send the confirmation code is sent according to the specified AccountRecoverySetting. For more information, see Recovering User Accounts in the *Amazon Cognito Developer Guide*. If neither a verified phone number nor a verified email exists, an `InvalidParameterException` is thrown. To use the confirmation code for resetting the password, call ConfirmForgotPassword.

> **Note**
> This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.
> If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In *sandbox mode* , you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see SMS message settings for Amazon Cognito user pools in the *Amazon Cognito Developer Guide*.

Request Syntax

```
{
   "AnalyticsMetadata": {
      "AnalyticsEndpointId": "string"
   },
   "ClientId": "string",
   "ClientMetadata": {
      "string" : "string"
   },
   "SecretHash": "string",
   "UserContextData": {
      "EncodedData": "string",
      "IpAddress": "string"
   },
   "Username": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 450).

The request accepts the following data in JSON format.

AnalyticsMetadata (p. 186)

The Amazon Pinpoint analytics metadata that contributes to your metrics for `ForgotPassword` calls.

Type: AnalyticsMetadataType (p. 358) object

Required: No

ClientId (p. 186)

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

ClientMetadata (p. 186)

A map of custom key-value pairs that you can provide as input for any custom workflows that this action triggers.

You create custom workflows by assigning AWS Lambda functions to user pool triggers. When you use the ForgotPassword API action, Amazon Cognito invokes any functions that are assigned to the following triggers: *pre sign-up*, *custom message*, and *user migration*. When Amazon Cognito invokes any of these functions, it passes a JSON payload, which the function receives as input. This payload contains a `clientMetadata` attribute, which provides the data that you assigned to the ClientMetadata parameter in your ForgotPassword request. In your function code in AWS Lambda, you can process the `clientMetadata` value to enhance your workflow for your specific needs.

For more information, see Customizing user pool Workflows with Lambda Triggers in the *Amazon Cognito Developer Guide*.

> **Note**
>
> When you use the ClientMetadata parameter, remember that Amazon Cognito won't do the following:
>
> - Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn't include triggers, the ClientMetadata parameter serves no purpose.
> - Validate the ClientMetadata value.
> - Encrypt the ClientMetadata value. Don't use Amazon Cognito to provide sensitive information.

Type: String to string map

Required: No

SecretHash (p. 186)

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. 186)

Contextual data about your user session, such as the device fingerprint, IP address, or location. Amazon Cognito advanced security evaluates the risk of an authentication event based on the context that your app generates and passes to Amazon Cognito when it makes API requests.

Type: UserContextDataType (p. 423) object

Required: No

Username (p. 186)

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. 188)

The code delivery details returned by the server in response to the request to reset a password.

Type: CodeDeliveryDetailsType (p. 365) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't allowed to use your email identity. HTTP status code: 400.

HTTP Status Code: 400

InvalidLambdaResponseException

This exception is thrown when Amazon Cognito 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 doesn't have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't 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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 Amazon Cognito encounters an unexpected exception with AWS Lambda.

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 isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

GetCSVHeader

Gets the header information for the comma-separated value (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. 450).

The request accepts the following data in JSON format.

UserPoolId (p. 191)

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. 191)

The header information of the CSV file for the user import job.

Type: Array of strings

UserPoolId (p. 191)

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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

AccessToken (p. 193)

A valid access token that Amazon Cognito issued to the user whose device information you want to request.

Type: String

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

Required: No

DeviceKey (p. 193)

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. 193)

The device.

Type: DeviceType (p. 374) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 not valid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't 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 can't 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 isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

GetGroup

Gets a group.

Calling this action 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. 450).

The request accepts the following data in JSON format.

GroupName (p. 196)

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. 196)

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. 196)

The group object for the group.

Type: GroupType (p. 383) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python

- AWS SDK for Ruby V3

GetIdentityProviderByIdentifier

Gets the specified IdP.

Request Syntax

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

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 450).

The request accepts the following data in JSON format.

IdpIdentifier (p. 199)

The IdP identifier.

Type: String

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

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

Required: Yes

UserPoolId (p. 199)

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. 199)

The IdP object.

Type: IdentityProviderType (p. 386) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

UserPoolId (p. 202)

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. 202)

The signing certificate.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

GetUICustomization

Gets the user interface (UI) Customization information for a particular app client's app UI, if any such information exists for the client. If nothing is set for the particular client, but there is an existing pool level customization (the app `clientId` is `ALL`), then that information 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. 450).

The request accepts the following data in JSON format.

ClientId (p. 204)

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. 204)

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. 204)

The UI customization information.

Type: UICustomizationType (p. 420) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript

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

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. 450).

The request accepts the following data in JSON format.

AccessToken (p. 207)

A non-expired access token for the user whose information you want to query.

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. 207)

This response parameter is no longer supported. It provides information only about SMS MFA configurations. It doesn't provide information about time-based one-time password (TOTP) software token MFA configurations. To look up information about either type of MFA configuration, use UserMFASettingList instead.

Type: Array of MFAOptionType (p. 393) objects

PreferredMfaSetting (p. 207)

The user's preferred MFA setting.

Type: String

UserAttributes (p. 207)

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. 359) objects

UserMFASettingList (p. 207)

The MFA options that are activated for the user. The possible values in this list are `SMS_MFA` and `SOFTWARE_TOKEN_MFA`.

Type: Array of strings

Username (p. 207)

The user name of the user you want 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. 452).

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 isn't 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 can't 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 isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

GetUserAttributeVerificationCode

Generates a user attribute verification code for the specified attribute name. Sends a message to a user with a code that they must return in a VerifyUserAttribute request.

Note

This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.

If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In *sandbox mode* , you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see SMS message settings for Amazon Cognito user pools in the *Amazon Cognito Developer Guide*.

Request Syntax

```
{
   "AccessToken": "string",
   "AttributeName": "string",
   "ClientMetadata": {
      "string" : "string"
   }
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 450).

The request accepts the following data in JSON format.

AccessToken (p. 210)

A non-expired access token for the user whose attribute verification code you want to generate.

Type: String

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

Required: Yes

AttributeName (p. 210)

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

ClientMetadata (p. 210)

A map of custom key-value pairs that you can provide as input for any custom workflows that this action triggers.

You create custom workflows by assigning AWS Lambda functions to user pool triggers. When you use the GetUserAttributeVerificationCode API action, Amazon Cognito invokes the function that is assigned to the *custom message* trigger. When Amazon Cognito invokes this function, it passes a JSON payload, which the function receives as input. This payload contains a `clientMetadata` attribute, which provides the data that you assigned to the ClientMetadata parameter in your GetUserAttributeVerificationCode request. In your function code in AWS Lambda, you can process the `clientMetadata` value to enhance your workflow for your specific needs.

For more information, see Customizing user pool Workflows with Lambda Triggers in the *Amazon Cognito Developer Guide*.

> **Note**
>
> When you use the ClientMetadata parameter, remember that Amazon Cognito won't do the following:
>
> - Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn't include triggers, the ClientMetadata parameter serves no purpose.
> - Validate the ClientMetadata value.
> - Encrypt the ClientMetadata value. Don't use Amazon Cognito to provide sensitive information.

Type: String to string map

Required: No

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. 211)

The code delivery details returned by the server in response to the request to get the user attribute verification code.

Type: CodeDeliveryDetailsType (p. 365) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't allowed to use your email identity. HTTP status code: 400.

HTTP Status Code: 400

InvalidLambdaResponseException

This exception is thrown when Amazon Cognito 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 doesn't have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't 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 isn't 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 can't 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 Amazon Cognito encounters an unexpected exception with AWS Lambda.

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 isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

UserPoolId (p. 214)

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",
         "SnsRegion": "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. 214)

The multi-factor (MFA) configuration. Valid values include:

- `OFF` MFA won't be used for any users.
- `ON` MFA is required for all users to sign in.
- `OPTIONAL` MFA will be required only for individual users who have an MFA factor activated.

Type: String

Valid Values: `OFF | ON | OPTIONAL`

SmsMfaConfiguration (p. 214)

The SMS text message multi-factor (MFA) configuration.

Type: SmsMfaConfigType (p. 414) object

SoftwareTokenMfaConfiguration (p. 214)

The software token multi-factor (MFA) configuration.

Type: SoftwareTokenMfaConfigType (p. 416) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

GlobalSignOut

Signs out users from all devices. It also invalidates all refresh tokens that Amazon Cognito has issued to a user. The user's current access and ID tokens remain valid until their expiry. By default, access and ID tokens expire one hour after Amazon Cognito issues them. A user can still use a hosted UI cookie to retrieve new tokens for the duration of the cookie validity period of 1 hour.

Request Syntax

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

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 450).

The request accepts the following data in JSON format.

AccessToken (p. 217)

A valid access token that Amazon Cognito issued to the user who you want to sign out.

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. 452).

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 isn't 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 can't 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 isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

InitiateAuth

Initiates sign-in for a user in the Amazon Cognito user directory. You can't sign in a user with a federated IdP with `InitiateAuth`. For more information, see Adding user pool sign-in through a third party.

Note

This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.

If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In *sandbox mode* , you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see SMS message settings for Amazon Cognito user pools in the *Amazon Cognito Developer Guide*.

Request Syntax

```
{
   "AnalyticsMetadata": {
      "AnalyticsEndpointId": "string"
   },
   "AuthFlow": "string",
   "AuthParameters": {
      "string" : "string"
   },
   "ClientId": "string",
   "ClientMetadata": {
      "string" : "string"
   },
   "UserContextData": {
      "EncodedData": "string",
      "IpAddress": "string"
   }
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 450).

The request accepts the following data in JSON format.

AnalyticsMetadata (p. 219)

The Amazon Pinpoint analytics metadata that contributes to your metrics for `InitiateAuth` calls.

Type: AnalyticsMetadataType (p. 358) object

Required: No

AuthFlow (p. 219)

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

- `REFRESH_TOKEN_AUTH` takes in a valid refresh token and returns new tokens.

- USER_SRP_AUTH takes in USERNAME and SRP_A and returns the SRP variables to be used for next challenge execution.
- USER_PASSWORD_AUTH takes in USERNAME and PASSWORD and returns 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; user name and password are passed directly. If a user migration Lambda trigger is set, this flow will invoke the user migration Lambda if it doesn't find the user name in the user pool.

ADMIN_NO_SRP_AUTH isn't a valid value.

Type: String

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

Required: Yes

AuthParameters (p. 219)

The authentication parameters. These are inputs corresponding to the AuthFlow that you're 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. To start the authentication flow with password verification, include ChallengeName: SRP_A and SRP_A: (The SRP_A Value).

Type: String to string map

Required: No

ClientId (p. 219)

The app client ID.

Type: String

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

Pattern: [\w+]+

Required: Yes

ClientMetadata (p. 219)

A map of custom key-value pairs that you can provide as input for certain custom workflows that this action triggers.

You create custom workflows by assigning AWS Lambda functions to user pool triggers. When you use the InitiateAuth API action, Amazon Cognito invokes the Lambda functions that are specified for various triggers. The ClientMetadata value is passed as input to the functions for only the following triggers:

- Pre signup
- Pre authentication
- User migration

When Amazon Cognito invokes the functions for these triggers, it passes a JSON payload, which the function receives as input. This payload contains a `validationData` attribute, which provides the data that you assigned to the ClientMetadata parameter in your InitiateAuth request. In your function code in Lambda, you can process the `validationData` value to enhance your workflow for your specific needs.

When you use the InitiateAuth API action, Amazon Cognito also invokes the functions for the following triggers, but it doesn't provide the ClientMetadata value as input:

- Post authentication
- Custom message
- Pre token generation
- Create auth challenge
- Define auth challenge
- Verify auth challenge

For more information, see Customizing user pool Workflows with Lambda Triggers in the *Amazon Cognito Developer Guide*.

> **Note**
>
> When you use the ClientMetadata parameter, remember that Amazon Cognito won't do the following:
>
> - Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn't include triggers, the ClientMetadata parameter serves no purpose.
> - Validate the ClientMetadata value.
> - Encrypt the ClientMetadata value. Don't use Amazon Cognito to provide sensitive information.

Type: String to string map

Required: No

UserContextData (p. 219)

Contextual data about your user session, such as the device fingerprint, IP address, or location. Amazon Cognito advanced security evaluates the risk of an authentication event based on the context that your app generates and passes to Amazon Cognito when it makes API requests.

Type: UserContextDataType (p. 423) 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. 221)

The result of the authentication response. This result is only returned if the caller doesn't 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. 360) object

ChallengeName (p. 221)

The name of the challenge that you're responding to with this call. This name is returned in the `AdminInitiateAuth` response if you must pass another challenge.

Valid values include the following:

> **Note**
> All of the following 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 activated 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 who are required to change their passwords after successful first login.

  Respond to this challenge with `NEW_PASSWORD` and any required attributes that Amazon Cognito returned in the `requiredAttributes` parameter. You can also set values for attributes that aren't required by your user pool and that your app client can write. For more information, see RespondToAuthChallenge.

  > **Note**
  > In a `NEW_PASSWORD_REQUIRED` challenge response, you can't modify a required attribute that already has a value. In `RespondToAuthChallenge`, set a value for any keys that Amazon Cognito returned in the `requiredAttributes` parameter, then use the `UpdateUserAttributes` API operation to modify the value of any additional attributes.

- MFA_SETUP: For users who are required to setup an MFA factor before they can sign in. The MFA types activated for the user pool will be listed in the challenge parameters MFA_CAN_SETUP value.

  To set up software token MFA, use the session returned here from InitiateAuth as an input to AssociateSoftwareToken. Use the session returned by VerifySoftwareToken as an input to RespondToAuthChallenge with challenge name MFA_SETUP to complete sign-in. To set up SMS MFA, an administrator should help the user to add a phone number to their account, and then the user should call InitiateAuth again to restart sign-in.

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. 221)

The challenge parameters. These are returned in the InitiateAuth response if you must 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. 221)

The session that should pass both ways in challenge-response calls to the service. If the caller must pass 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. 452).

InternalErrorException

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

HTTP Status Code: 500

InvalidLambdaResponseException

This exception is thrown when Amazon Cognito 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 doesn't have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't 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 not valid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't 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 can't 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 Amazon Cognito encounters an unexpected exception with AWS Lambda.

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 isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

ListDevices

Lists the sign-in devices that Amazon Cognito has registered to the current user.

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. 450).

The request accepts the following data in JSON format.

AccessToken (p. 226)

A valid access token that Amazon Cognito issued to the user whose list of devices you want to view.

Type: String

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

Required: Yes

Limit (p. 226)

The limit of the device request.

Type: Integer

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

Required: No

PaginationToken (p. 226)

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. 226)

The devices returned in the list devices response.

Type: Array of DeviceType (p. 374) objects

PaginationToken (p. 226)

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. 452).

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 not valid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't 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 can't 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 isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

ListGroups

Lists the groups associated with a user pool.

Calling this action 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. 450).

The request accepts the following data in JSON format.

Limit (p. 229)

The limit of the request to list groups.

Type: Integer

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

Required: No

NextToken (p. 229)

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. 229)

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. 229)

The group objects for the groups.

Type: Array of GroupType (p. 383) objects

NextToken (p. 229)

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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

ListIdentityProviders

Lists information about all IdPs 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. 450).

The request accepts the following data in JSON format.

MaxResults (p. 232)

The maximum number of IdPs to return.

Type: Integer

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

Required: No

NextToken (p. 232)

A pagination token.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

Required: No

UserPoolId (p. 232)

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. 232)

A pagination token.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

Providers (p. 232)

A list of IdP objects.

Type: Array of ProviderDescription (p. 401) 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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

MaxResults (p. 235)

The maximum number of resource servers to return.

Type: Integer

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

Required: No

NextToken (p. 235)

A pagination token.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

Required: No

UserPoolId (p. 235)

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. 235)

A pagination token.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

ResourceServers (p. 235)

The resource servers.

Type: Array of ResourceServerType (p. 405) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

ListTagsForResource

Lists the tags that are assigned to an Amazon Cognito user pool.

A tag is a label that you can apply to user pools to categorize and manage them in different ways, such as by purpose, owner, environment, or other criteria.

You can use this action up to 10 times per second, per account.

Request Syntax

```
{
   "ResourceArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 450).

The request accepts the following data in JSON format.

ResourceArn (p. 238)

The Amazon Resource Name (ARN) of the user pool that the tags are assigned to.

Type: String

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

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

Required: Yes

Response Syntax

```
{
   "Tags": {
      "string" : "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.

Tags (p. 238)

The tags that are assigned to the user pool.

Type: String to string map

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

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

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

MaxResults (p. 240)

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. 240)

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. 240)

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. 240)

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. 240)

The user import jobs.

Type: Array of UserImportJobType (p. 424) 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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

MaxResults (p. 243)

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. 243)

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. 243)

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. 243)

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. 243)

The user pool clients in the response that lists user pool clients.

Type: Array of UserPoolClientDescription (p. 429) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

MaxResults (p. 246)

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. 246)

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",
            "CustomEmailSender": {
               "LambdaArn": "string",
               "LambdaVersion": "string"
            },
            "CustomMessage": "string",
            "CustomSMSSender": {
               "LambdaArn": "string",
               "LambdaVersion": "string"
            },
            "DefineAuthChallenge": "string",
            "KMSKeyID": "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. 246)

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. 246)

The user pools from the response to list users.

Type: Array of UserPoolDescriptionType (p. 437) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

AttributesToGet (p. 249)

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. 249)

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 aren't searchable.

> **Note**
> You can also list users with a client-side filter. The server-side filter matches no more than one attribute. For an advanced search, use a client-side filter with the `--query` parameter of the `list-users` action in the AWS CLI. When you use a client-side filter, ListUsers returns a paginated list of zero or more users. You can receive multiple pages in a row with zero results. Repeat the query with each pagination token that is returned until you receive a null pagination token value, and then review the combined result.
> For more information about server-side and client-side filtering, see FilteringAWS CLI output in the AWS Command Line Interface User Guide.

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. 249)

Maximum number of users to be returned.

Type: Integer

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

Required: No

PaginationToken (p. 249)

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. 249)

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. 250)

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. 250)

The users returned in the request to list users.

Type: Array of UserType (p. 446) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

ListUsersInGroup

Lists the users in the specified group.

Calling this action 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. 450).

The request accepts the following data in JSON format.

GroupName (p. 253)

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. 253)

The limit of the request to list users.

Type: Integer

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

Required: No

NextToken (p. 253)

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. 253)

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. 254)

An identifier that you can use in a later request to return the next set of items in the list.

Type: String

Length Constraints: Minimum length of 1.

Pattern: `[\S]+`

Users (p. 254)

The users returned in the request to list users.

Type: Array of UserType (p. 446) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

ResendConfirmationCode

Resends the confirmation (for confirmation of registration) to a specific user in the user pool.

> **Note**
> This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.
> If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In *sandbox mode* , you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see SMS message settings for Amazon Cognito user pools in the *Amazon Cognito Developer Guide*.

Request Syntax

```
{
   "AnalyticsMetadata": {
      "AnalyticsEndpointId": "string"
   },
   "ClientId": "string",
   "ClientMetadata": {
      "string" : "string"
   },
   "SecretHash": "string",
   "UserContextData": {
      "EncodedData": "string",
      "IpAddress": "string"
   },
   "Username": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 450).

The request accepts the following data in JSON format.

AnalyticsMetadata (p. 256)

The Amazon Pinpoint analytics metadata that contributes to your metrics for `ResendConfirmationCode` calls.

Type: AnalyticsMetadataType (p. 358) object

Required: No

ClientId (p. 256)

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

ClientMetadata (p. 256)

A map of custom key-value pairs that you can provide as input for any custom workflows that this action triggers.

You create custom workflows by assigning AWS Lambda functions to user pool triggers. When you use the ResendConfirmationCode API action, Amazon Cognito invokes the function that is assigned to the *custom message* trigger. When Amazon Cognito invokes this function, it passes a JSON payload, which the function receives as input. This payload contains a `clientMetadata` attribute, which provides the data that you assigned to the ClientMetadata parameter in your ResendConfirmationCode request. In your function code in Lambda, you can process the `clientMetadata` value to enhance your workflow for your specific needs.

For more information, see Customizing user pool Workflows with Lambda Triggers in the *Amazon Cognito Developer Guide*.

Note

When you use the ClientMetadata parameter, remember that Amazon Cognito won't do the following:

- Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn't include triggers, the ClientMetadata parameter serves no purpose.
- Validate the ClientMetadata value.
- Encrypt the ClientMetadata value. Don't use Amazon Cognito to provide sensitive information.

Type: String to string map

Required: No

SecretHash (p. 256)

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. 256)

Contextual data about your user session, such as the device fingerprint, IP address, or location. Amazon Cognito advanced security evaluates the risk of an authentication event based on the context that your app generates and passes to Amazon Cognito when it makes API requests.

Type: UserContextDataType (p. 423) object

Required: No

Username (p. 256)

The `username` attribute of the user to whom you want 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. 258)

The code delivery details returned by the server in response to the request to resend the confirmation code.

Type: CodeDeliveryDetailsType (p. 365) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't allowed to use your email identity. HTTP status code: 400.

HTTP Status Code: 400

InvalidLambdaResponseException

This exception is thrown when Amazon Cognito 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 doesn't have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't 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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 Amazon Cognito encounters an unexpected exception with AWS Lambda.

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 isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

RespondToAuthChallenge

Responds to the authentication challenge.

> **Note**
> This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.
> If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In *sandbox mode* , you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see SMS message settings for Amazon Cognito user pools in the *Amazon Cognito Developer Guide*.

Request Syntax

```
{
   "AnalyticsMetadata": {
      "AnalyticsEndpointId": "string"
   },
   "ChallengeName": "string",
   "ChallengeResponses": {
      "string" : "string"
   },
   "ClientId": "string",
   "ClientMetadata": {
      "string" : "string"
   },
   "Session": "string",
   "UserContextData": {
      "EncodedData": "string",
      "IpAddress": "string"
   }
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 450).

The request accepts the following data in JSON format.

AnalyticsMetadata (p. 261)

The Amazon Pinpoint analytics metadata that contributes to your metrics for `RespondToAuthChallenge` calls.

Type: AnalyticsMetadataType (p. 358) object

Required: No

ChallengeName (p. 261)

The challenge name. For more information, see InitiateAuth.

ADMIN_NO_SRP_AUTH isn't 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. 261)

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

> **Note**
>
> SECRET_HASH (if app client is configured with client secret) applies to all of the inputs that follow (including SOFTWARE_TOKEN_MFA).

- SMS_MFA: SMS_MFA_CODE, USERNAME.
- PASSWORD_VERIFIER: PASSWORD_CLAIM_SIGNATURE, PASSWORD_CLAIM_SECRET_BLOCK, TIMESTAMP, USERNAME.

  > **Note**
  >
  > PASSWORD_VERIFIER requires DEVICE_KEY when you sign in with a remembered device.
- NEW_PASSWORD_REQUIRED: NEW_PASSWORD, USERNAME, SECRET_HASH (if app client is configured with client secret). To set any required attributes that Amazon Cognito returned as requiredAttributes in the InitiateAuth response, add a userAttributes.*attributename* parameter. This parameter can also set values for writable attributes that aren't required by your user pool.

  > **Note**
  >
  > In a NEW_PASSWORD_REQUIRED challenge response, you can't modify a required attribute that already has a value. In RespondToAuthChallenge, set a value for any keys that Amazon Cognito returned in the requiredAttributes parameter, then use the UpdateUserAttributes API operation to modify the value of any additional attributes.
- SOFTWARE_TOKEN_MFA: USERNAME and SOFTWARE_TOKEN_MFA_CODE are required attributes.
- DEVICE_SRP_AUTH requires USERNAME, DEVICE_KEY, SRP_A (and SECRET_HASH).
- DEVICE_PASSWORD_VERIFIER requires everything that PASSWORD_VERIFIER requires, plus DEVICE_KEY.
- MFA_SETUP requires USERNAME, plus you must use the session value returned by VerifySoftwareToken in the Session parameter.

Type: String to string map

Required: No

ClientId (p. 261)

The app client ID.

Type: String

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

Pattern: `[\w+]+`

Required: Yes

ClientMetadata (p. 261)

A map of custom key-value pairs that you can provide as input for any custom workflows that this action triggers.

You create custom workflows by assigning AWS Lambda functions to user pool triggers. When you use the RespondToAuthChallenge API action, Amazon Cognito invokes any functions that are assigned to the following triggers: *post authentication*, *pre token generation*, *define auth challenge*, *create auth challenge*, and *verify auth challenge*. When Amazon Cognito invokes any of these functions, it passes a JSON payload, which the function receives as input. This payload contains a `clientMetadata` attribute, which provides the data that you assigned to the ClientMetadata parameter in your RespondToAuthChallenge request. In your function code in AWS Lambda, you can process the `clientMetadata` value to enhance your workflow for your specific needs.

For more information, see Customizing user pool Workflows with Lambda Triggers in the *Amazon Cognito Developer Guide*.

Note

When you use the ClientMetadata parameter, remember that Amazon Cognito won't do the following:

- Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn't include triggers, the ClientMetadata parameter serves no purpose.
- Validate the ClientMetadata value.
- Encrypt the ClientMetadata value. Don't use Amazon Cognito to provide sensitive information.

Type: String to string map

Required: No

Session (p. 261)

The session that should be passed both ways in challenge-response calls to the service. If `InitiateAuth` or `RespondToAuthChallenge` API call determines that the caller must pass 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. 261)

Contextual data about your user session, such as the device fingerprint, IP address, or location. Amazon Cognito advanced security evaluates the risk of an authentication event based on the context that your app generates and passes to Amazon Cognito when it makes API requests.

Type: UserContextDataType (p. 423) 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. 263)

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

Type: AuthenticationResultType (p. 360) object

ChallengeName (p. 263)

The challenge name. For more information, see InitiateAuth.

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. 263)

The challenge parameters. For more information, see InitiateAuth.

Type: String to string map

Session (p. 263)

The session that should be passed both ways in challenge-response calls to the service. If the caller must pass 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. 452).

AliasExistsException

This exception is thrown when a user tries to confirm the account with an email address or phone number that has already been supplied as an alias for a different user profile. This exception indicates that an account with this email address or phone already exists in a user pool that you've configured to use email address or phone number as a sign-in alias.

HTTP Status Code: 400

CodeMismatchException

This exception is thrown if the provided code doesn't 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 Amazon Cognito 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 Amazon Cognito encounters an invalid password.

HTTP Status Code: 400

InvalidSmsRoleAccessPolicyException

This exception is returned when the role provided for SMS configuration doesn't have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't 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 not valid.

HTTP Status Code: 400

MFAMethodNotFoundException

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

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't 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 can't find the requested resource.

HTTP Status Code: 400

SoftwareTokenMFANotFoundException

This exception is thrown when the software token time-based one-time password (TOTP) multi-factor authentication (MFA) isn't activated 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 Amazon Cognito encounters an unexpected exception with AWS Lambda.

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 isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't 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 V2
- AWS SDK for JavaScript

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

RevokeToken

Revokes all of the access tokens generated by the specified refresh token. After the token is revoked, you can't use the revoked token to access Amazon Cognito authenticated APIs.

Request Syntax

```
{
   "ClientId": "string",
   "ClientSecret": "string",
   "Token": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 450).

The request accepts the following data in JSON format.

ClientId (p. 268)

The client ID for the token that you want to revoke.

Type: String

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

Pattern: `[\w+]+`

Required: Yes

ClientSecret (p. 268)

The secret for the client ID. This is required only if the client ID has a secret.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `[\w+]+`

Required: No

Token (p. 268)

The refresh token that you want to revoke.

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. 452).

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

TooManyRequestsException

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

HTTP Status Code: 400

UnauthorizedException

Exception that is thrown when the request isn't authorized. This can happen due to an invalid access token in the request.

HTTP Status Code: 400

UnsupportedOperationException

Exception that is thrown when you attempt to perform an operation that isn't enabled for the user pool client.

HTTP Status Code: 400

UnsupportedTokenTypeException

Exception that is thrown when an unsupported token is passed to an 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

SetRiskConfiguration

Configures actions on detected risks. To delete the risk configuration for `UserPoolId` or `ClientId`, pass null values for all four configuration types.

To activate Amazon Cognito advanced security features, update the user pool to include the `UserPoolAddOns` key`AdvancedSecurityMode`.

See UpdateUserPool (p. 323).

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. 450).

The request accepts the following data in JSON format.

AccountTakeoverRiskConfiguration (p. 270)

The account takeover risk configuration.

Type: AccountTakeoverRiskConfigurationType (p. 354) object

Required: No

ClientId (p. 270)

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 isn't 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. 270)

The compromised credentials risk configuration.

Type: CompromisedCredentialsRiskConfigurationType (p. 367) object

Required: No

RiskExceptionConfiguration (p. 270)

The configuration to override the risk decision.

Type: RiskExceptionConfigurationType (p. 409) object

Required: No

UserPoolId (p. 270)

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. 271)

The risk configuration.

Type: RiskConfigurationType (p. 407) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't 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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 aren't 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 V2

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

SetUICustomization

Sets the user interface (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 is used for every client that has no previously set UI customization. If you specify UI customization settings for a particular client, it will no longer return 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. 450).

The request accepts the following data in JSON format.

ClientId (p. 275)

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. 275)

The CSS values in the UI customization.

Type: String

Required: No

ImageFile (p. 275)

The uploaded logo image for the UI customization.

Type: Base64-encoded binary data object

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

```
{
   "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. 276)

The UI customization information.

Type: UICustomizationType (p. 420) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

SetUserMFAPreference

Set the user's multi-factor authentication (MFA) method preference, including which MFA factors are activated and if any are preferred. Only one factor can be set as preferred. The preferred MFA factor will be used to authenticate a user if multiple factors are activated. If multiple options are activated and no preference is set, a challenge to choose an MFA option will be returned during sign-in. If an MFA type is activated for a user, the user will be prompted for MFA during all sign-in attempts unless device tracking is turned on and the device has been trusted. If you want MFA to be applied selectively based on the assessed risk level of sign-in attempts, deactivate MFA for users and turn on Adaptive Authentication for the user pool.

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. 450).

The request accepts the following data in JSON format.

AccessToken (p. 278)

A valid access token that Amazon Cognito issued to the user whose MFA preference you want to set.

Type: String

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

Required: Yes

SMSMfaSettings (p. 278)

The SMS text message multi-factor authentication (MFA) settings.

Type: SMSMfaSettingsType (p. 415) object

Required: No

SoftwareTokenMfaSettings (p. 278)

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

Type: SoftwareTokenMfaSettingsType (p. 417) 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. 452).

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 isn't 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 can't find the requested resource.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't 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 V2
- AWS SDK for JavaScript

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

SetUserPoolMfaConfig

Sets the user pool multi-factor authentication (MFA) configuration.

> **Note**
> This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.
> If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In *sandbox mode* , you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see SMS message settings for Amazon Cognito user pools in the *Amazon Cognito Developer Guide*.

Request Syntax

```
{
   "MfaConfiguration": "string",
   "SmsMfaConfiguration": {
      "SmsAuthenticationMessage": "string",
      "SmsConfiguration": {
         "ExternalId": "string",
         "SnsCallerArn": "string",
         "SnsRegion": "string"
      }
   },
   "SoftwareTokenMfaConfiguration": {
      "Enabled": boolean
   },
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 450).

The request accepts the following data in JSON format.

MfaConfiguration (p. 281)

The MFA configuration. If you set the MfaConfiguration value to ‘ON’, only users who have set up an MFA factor can sign in. To learn more, see Adding Multi-Factor Authentication (MFA) to a user pool. Valid values include:

- `OFF` MFA won't be used for any users.
- `ON` MFA is required for all users to sign in.
- `OPTIONAL` MFA will be required only for individual users who have an MFA factor activated.

Type: String

Valid Values: `OFF | ON | OPTIONAL`

Required: No

SmsMfaConfiguration (p. 281)

The SMS text message MFA configuration.

Type: SmsMfaConfigType (p. 414) object

Required: No

SoftwareTokenMfaConfiguration (p. 281)

The software token MFA configuration.

Type: SoftwareTokenMfaConfigType (p. 416) object

Required: No

UserPoolId (p. 281)

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",
         "SnsRegion": "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. 282)

The MFA configuration. Valid values include:

- `OFF` MFA won't be used for any users.
- `ON` MFA is required for all users to sign in.
- `OPTIONAL` MFA will be required only for individual users who have an MFA factor enabled.

Type: String

Valid Values: `OFF | ON | OPTIONAL`

SmsMfaConfiguration (p. 282)

The SMS text message MFA configuration.

Type: SmsMfaConfigType (p. 414) object

SoftwareTokenMfaConfiguration (p. 282)

The software token MFA configuration.

Type: SoftwareTokenMfaConfigType (p. 416) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 doesn't have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't 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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

SetUserSettings

This action is no longer supported. You can use it to configure only SMS MFA. You can't use it to configure time-based one-time password (TOTP) software token MFA. To configure either type of MFA, use SetUserMFAPreference instead.

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. 450).

The request accepts the following data in JSON format.

AccessToken (p. 285)

A valid access token that Amazon Cognito issued to the user whose user settings you want to configure.

Type: String

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

Required: Yes

MFAOptions (p. 285)

You can use this parameter only to set an SMS configuration that uses SMS for delivery.

Type: Array of MFAOptionType (p. 393) 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. 452).

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 isn't 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 can't find the requested resource.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

SignUp

Registers the user in the specified user pool and creates a user name, password, and user attributes.

> **Note**
> This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.
> If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In *sandbox mode* , you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see SMS message settings for Amazon Cognito user pools in the *Amazon Cognito Developer Guide*.

Request Syntax

```
{
   "AnalyticsMetadata": {
      "AnalyticsEndpointId": "string"
   },
   "ClientId": "string",
   "ClientMetadata": {
      "string" : "string"
   },
   "Password": "string",
   "SecretHash": "string",
   "UserAttributes": [
      {
         "Name": "string",
         "Value": "string"
      }
   ],
   "UserContextData": {
      "EncodedData": "string",
      "IpAddress": "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. 450).

The request accepts the following data in JSON format.

AnalyticsMetadata (p. 287)

The Amazon Pinpoint analytics metadata that contributes to your metrics for `SignUp` calls.

Type: AnalyticsMetadataType (p. 358) object

Required: No

ClientId (p. 287)

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

ClientMetadata (p. 287)

A map of custom key-value pairs that you can provide as input for any custom workflows that this action triggers.

You create custom workflows by assigning AWS Lambda functions to user pool triggers. When you use the SignUp API action, Amazon Cognito invokes any functions that are assigned to the following triggers: *pre sign-up*, *custom message*, and *post confirmation*. When Amazon Cognito invokes any of these functions, it passes a JSON payload, which the function receives as input. This payload contains a `clientMetadata` attribute, which provides the data that you assigned to the ClientMetadata parameter in your SignUp request. In your function code in Lambda, you can process the `clientMetadata` value to enhance your workflow for your specific needs.

For more information, see Customizing user pool Workflows with Lambda Triggers in the *Amazon Cognito Developer Guide*.

Note

When you use the ClientMetadata parameter, remember that Amazon Cognito won't do the following:

- Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn't include triggers, the ClientMetadata parameter serves no purpose.
- Validate the ClientMetadata value.
- Encrypt the ClientMetadata value. Don't use Amazon Cognito to provide sensitive information.

Type: String to string map

Required: No

Password (p. 287)

The password of the user you want to register.

Type: String

Length Constraints: Maximum length of 256.

Pattern: `[\S]+`

Required: Yes

SecretHash (p. 287)

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. 287)

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. 359) objects

Required: No

UserContextData (p. 287)

Contextual data about your user session, such as the device fingerprint, IP address, or location. Amazon Cognito advanced security evaluates the risk of an authentication event based on the context that your app generates and passes to Amazon Cognito when it makes API requests.

Type: UserContextDataType (p. 423) object

Required: No

Username (p. 287)

The user name of the user you want 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. 287)

The validation data in the request to register a user.

Type: Array of AttributeType (p. 359) 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. 289)

The code delivery details returned by the server response to the user registration request.

Type: CodeDeliveryDetailsType (p. 365) object

UserConfirmed (p. 289)

A response from the server indicating that a user registration has been confirmed.

Type: Boolean

UserSub (p. 289)

The UUID of the authenticated user. This isn't the same as `username`.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't allowed to use your email identity. HTTP status code: 400.

HTTP Status Code: 400

InvalidLambdaResponseException

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

HTTP Status Code: 400

InvalidSmsRoleAccessPolicyException

This exception is returned when the role provided for SMS configuration doesn't have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't 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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 Amazon Cognito encounters an unexpected exception with AWS Lambda.

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

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. 450).

The request accepts the following data in JSON format.

JobId (p. 293)

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. 293)

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. 293)

The job object that represents the user import job.

Type: UserImportJobType (p. 424) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't 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 can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

JobId (p. 296)

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. 296)

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. 296)

The job object that represents the user import job.

Type: UserImportJobType (p. 424) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't 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 can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

TagResource

Assigns a set of tags to an Amazon Cognito user pool. A tag is a label that you can use to categorize and manage user pools in different ways, such as by purpose, owner, environment, or other criteria.

Each tag consists of a key and value, both of which you define. A key is a general category for more specific values. For example, if you have two versions of a user pool, one for testing and another for production, you might assign an `Environment` tag key to both user pools. The value of this key might be `Test` for one user pool, and `Production` for the other.

Tags are useful for cost tracking and access control. You can activate your tags so that they appear on the Billing and Cost Management console, where you can track the costs associated with your user pools. In an AWS Identity and Access Management policy, you can constrain permissions for user pools based on specific tags or tag values.

You can use this action up to 5 times per second, per account. A user pool can have as many as 50 tags.

Request Syntax

```
{
   "ResourceArn": "string",
   "Tags": {
      "string" : "string"
   }
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 450).

The request accepts the following data in JSON format.

ResourceArn (p. 299)

The Amazon Resource Name (ARN) of the user pool to assign the tags to.

Type: String

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

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

Required: Yes

Tags (p. 299)

The tags to assign to the user pool.

Type: String to string map

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

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

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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UntagResource

Removes the specified tags from an Amazon Cognito user pool. You can use this action up to 5 times per second, per account.

Request Syntax

```
{
   "ResourceArn": "string",
   "TagKeys": [ "string" ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 450).

The request accepts the following data in JSON format.

ResourceArn (p. 301)

The Amazon Resource Name (ARN) of the user pool that the tags are assigned to.

Type: String

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

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

Required: Yes

TagKeys (p. 301)

The keys of the tags to remove from the user pool.

Type: Array of strings

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

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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

EventId (p. 303)

The event ID.

Type: String

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

Pattern: `[\w+-]+`

Required: Yes

FeedbackToken (p. 303)

The feedback token.

Type: String

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

Required: Yes

FeedbackValue (p. 303)

The authentication event feedback value.

Type: String

Valid Values: `Valid | Invalid`

Required: Yes

Username (p. 303)

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. 303)

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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 isn't found.

HTTP Status Code: 400

UserPoolAddOnNotEnabledException

This exception is thrown when user pool add-ons aren't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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. 450).

The request accepts the following data in JSON format.

AccessToken (p. 306)

A valid access token that Amazon Cognito issued to the user whose device status you want to update.

Type: String

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

Required: Yes

DeviceKey (p. 306)

The device key.

Type: String

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

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

Required: Yes

DeviceRememberedStatus (p. 306)

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. 452).

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 not valid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't 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 can't 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 isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateGroup

Updates the specified group with the specified attributes.

Calling this action 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. 450).

The request accepts the following data in JSON format.

Description (p. 309)

A string containing the new description of the group.

Type: String

Length Constraints: Maximum length of 2048.

Required: No

GroupName (p. 309)

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. 309)

The new precedence value for the group. For more information about this parameter, see CreateGroup.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

RoleArn (p. 309)

The new role Amazon Resource Name (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. 309)

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. 310)

The group object for the group.

Type: GroupType (p. 383) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateIdentityProvider

Updates IdP 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. 450).

The request accepts the following data in JSON format.

AttributeMapping (p. 312)

The IdP 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. 312)

A list of IdP 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. 312)

The IdP details to be updated, such as `MetadataURL` and `MetadataFile`.

Type: String to string map

Required: No

ProviderName (p. 312)

The IdP 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. 312)

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. 313)

The IdP object.

Type: IdentityProviderType (p. 386) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateResourceServer

Updates the name and scopes of resource server. All other fields are read-only.

> **Important**
> If you don't provide a value for an attribute, it is set to the default value.

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. 450).

The request accepts the following data in JSON format.

Identifier (p. 315)

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. 315)

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. 315)

The scope values to be set for the resource server.

Type: Array of ResourceServerScopeType (p. 404) objects

Array Members: Maximum number of 100 items.

Required: No

UserPoolId (p. 315)

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. 316)

The resource server.

Type: ResourceServerType (p. 405) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateUserAttributes

Allows a user to update a specific attribute (one at a time).

Note

This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.

If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In *sandbox mode* , you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see SMS message settings for Amazon Cognito user pools in the *Amazon Cognito Developer Guide*.

Request Syntax

```
{
   "AccessToken": "string",
   "ClientMetadata": {
      "string" : "string"
   },
   "UserAttributes": [
      {
         "Name": "string",
         "Value": "string"
      }
   ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 450).

The request accepts the following data in JSON format.

AccessToken (p. 318)

A valid access token that Amazon Cognito issued to the user whose user attributes you want to update.

Type: String

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

Required: Yes

ClientMetadata (p. 318)

A map of custom key-value pairs that you can provide as input for any custom workflows that this action initiates.

You create custom workflows by assigning Lambda functions to user pool triggers. When you use the UpdateUserAttributes API action, Amazon Cognito invokes the function that is assigned to the *custom message* trigger. When Amazon Cognito invokes this function, it passes a JSON payload, which the function receives as input. This payload contains a `clientMetadata` attribute, which provides the data that you assigned to the ClientMetadata parameter in your UpdateUserAttributes request. In your function code in Lambda, you can process the `clientMetadata` value to enhance your workflow for your specific needs.

For more information, see Customizing user pool Workflows with Lambda Triggers in the *Amazon Cognito Developer Guide*.

> **Note**
> When you use the ClientMetadata parameter, remember that Amazon Cognito won't do the following:
>
> - Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn't include triggers, the ClientMetadata parameter serves no purpose.
> - Validate the ClientMetadata value.
> - Encrypt the ClientMetadata value. Don't use Amazon Cognito to provide sensitive information.

Type: String to string map

Required: No

UserAttributes (p. 318)

An array of name-value pairs representing user attributes.

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

If you have set an attribute to require verification before Amazon Cognito updates its value, this request doesn’t immediately update the value of that attribute. After your user receives and responds to a verification message to verify the new value, Amazon Cognito updates the attribute value. Your user can sign in and receive messages with the original attribute value until they verify the new value.

Type: Array of AttributeType (p. 359) 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. 319)

The code delivery details list from the server for the request to update user attributes.

Type: Array of CodeDeliveryDetailsType (p. 365) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

AliasExistsException

This exception is thrown when a user tries to confirm the account with an email address or phone number that has already been supplied as an alias for a different user profile. This exception indicates that an account with this email address or phone already exists in a user pool that you've configured to use email address or phone number as a sign-in alias.

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 doesn't 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 isn't allowed to use your email identity. HTTP status code: 400.

HTTP Status Code: 400

InvalidLambdaResponseException

This exception is thrown when Amazon Cognito 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 doesn't have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't 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 isn't 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 can't 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 Amazon Cognito encounters an unexpected exception with AWS Lambda.

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 isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateUserPool

Updates the specified user pool with the specified attributes. You can get a list of the current user pool settings using DescribeUserPool. If you don't provide a value for an attribute, it will be set to the default value.

> **Note**
>
> This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.
>
> If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In *sandbox mode* , you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see SMS message settings for Amazon Cognito user pools in the *Amazon Cognito Developer Guide*.

Request Syntax

```
{
   "AccountRecoverySetting": {
      "RecoveryMechanisms": [
         {
            "Name": "string",
            "Priority": number
         }
      ]
   },
   "AdminCreateUserConfig": {
      "AllowAdminCreateUserOnly": boolean,
      "InviteMessageTemplate": {
         "EmailMessage": "string",
         "EmailSubject": "string",
         "SMSMessage": "string"
      },
      "UnusedAccountValidityDays": number
   },
   "AutoVerifiedAttributes": [ "string" ],
   "DeviceConfiguration": {
      "ChallengeRequiredOnNewDevice": boolean,
      "DeviceOnlyRememberedOnUserPrompt": boolean
   },
   "EmailConfiguration": {
      "ConfigurationSet": "string",
      "EmailSendingAccount": "string",
      "From": "string",
      "ReplyToEmailAddress": "string",
      "SourceArn": "string"
   },
   "EmailVerificationMessage": "string",
   "EmailVerificationSubject": "string",
   "LambdaConfig": {
      "CreateAuthChallenge": "string",
      "CustomEmailSender": {
         "LambdaArn": "string",
         "LambdaVersion": "string"
      },
      "CustomMessage": "string",
      "CustomSMSSender": {
         "LambdaArn": "string",
         "LambdaVersion": "string"
      },
      "DefineAuthChallenge": "string",
      "KMSKeyID": "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,
         "TemporaryPasswordValidityDays": number
      }
   },
   "SmsAuthenticationMessage": "string",
   "SmsConfiguration": {
      "ExternalId": "string",
      "SnsCallerArn": "string",
      "SnsRegion": "string"
   },
   "SmsVerificationMessage": "string",
   "UserAttributeUpdateSettings": {
      "AttributesRequireVerificationBeforeUpdate": [ "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. 450).

The request accepts the following data in JSON format.

AccountRecoverySetting (p. 323)

> The available verified method a user can use to recover their password when they call `ForgotPassword`. You can use this setting to define a preferred method when a user has more

than one method available. With this setting, SMS doesn't qualify for a valid password recovery mechanism if the user also has SMS multi-factor authentication (MFA) activated. In the absence of this setting, Amazon Cognito uses the legacy behavior to determine the recovery method where SMS is preferred through email.

Type: AccountRecoverySettingType (p. 351) object

Required: No

AdminCreateUserConfig (p. 323)

The configuration for `AdminCreateUser` requests.

Type: AdminCreateUserConfigType (p. 355) object

Required: No

AutoVerifiedAttributes (p. 323)

The attributes that are automatically verified when Amazon Cognito requests to update user pools.

Type: Array of strings

Valid Values: `phone_number | email`

Required: No

DeviceConfiguration (p. 323)

Device configuration.

Type: DeviceConfigurationType (p. 372) object

Required: No

EmailConfiguration (p. 323)

The email configuration of your user pool. The email configuration type sets your preferred sending method, AWS Region, and sender for email invitation and verification messages from your user pool.

Type: EmailConfigurationType (p. 377) object

Required: No

EmailVerificationMessage (p. 323)

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. 323)

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. 323)

The Lambda configuration information from the request to update the user pool.

Type: LambdaConfigType (p. 389) object

Required: No

MfaConfiguration (p. 323)

Possible values include:

- `OFF` - MFA tokens aren't required and can't be specified during user registration.
- `ON` - MFA tokens are required for all user registrations. You can only specify ON when you're initially creating a user pool. You can use the SetUserPoolMfaConfig API operation to turn MFA "ON" for existing user pools.
- `OPTIONAL` - Users have the option when registering to create an MFA token.

Type: String

Valid Values: `OFF | ON | OPTIONAL`

Required: No

Policies (p. 323)

A container with the policies you want to update in a user pool.

Type: UserPoolPolicyType (p. 439) object

Required: No

SmsAuthenticationMessage (p. 323)

The contents of the SMS authentication message.

Type: String

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

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

Required: No

SmsConfiguration (p. 323)

The SMS configuration with the settings that your Amazon Cognito user pool must use to send an SMS message from your AWS account through Amazon Simple Notification Service. To send SMS messages with Amazon SNS in the AWS Region that you want, the Amazon Cognito user pool uses an AWS Identity and Access Management (IAM) role in your AWS account.

Type: SmsConfigurationType (p. 412) object

Required: No

SmsVerificationMessage (p. 323)

A container with information about the SMS verification message.

Type: String

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

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

Required: No

UserAttributeUpdateSettings (p. 323)

The settings for updates to user attributes. These settings include the property `AttributesRequireVerificationBeforeUpdate`, a user-pool setting that tells Amazon Cognito how to handle changes to the value of your users' email address and phone number attributes. For more information, see Verifying updates to to email addresses and phone numbers.

Type: UserAttributeUpdateSettingsType (p. 422) object

Required: No

UserPoolAddOns (p. 323)

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

Type: UserPoolAddOnsType (p. 428) object

Required: No

UserPoolId (p. 323)

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. 323)

The tag keys and values to assign to the user pool. A tag is a label that you can use to categorize and manage user pools in different ways, such as by purpose, owner, environment, or other criteria.

Type: String to string map

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

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

Required: No

VerificationMessageTemplate (p. 323)

The template for verification messages.

Type: VerificationMessageTemplateType (p. 448) 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. 452).

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 isn't 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 doesn't have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't 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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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're 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 can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateUserPoolClient

Updates the specified user pool app client with the specified attributes. You can get a list of the current user pool app client settings using DescribeUserPoolClient.

> **Important**
> If you don't provide a value for an attribute, it will be set to the default value.

You can also use this operation to enable token revocation for user pool clients. For more information about revoking tokens, see RevokeToken.

Request Syntax

```
{
   "AccessTokenValidity": number,
   "AllowedOAuthFlows": [ "string" ],
   "AllowedOAuthFlowsUserPoolClient": boolean,
   "AllowedOAuthScopes": [ "string" ],
   "AnalyticsConfiguration": {
      "ApplicationArn": "string",
      "ApplicationId": "string",
      "ExternalId": "string",
      "RoleArn": "string",
      "UserDataShared": boolean
   },
   "CallbackURLs": [ "string" ],
   "ClientId": "string",
   "ClientName": "string",
   "DefaultRedirectURI": "string",
   "EnablePropagateAdditionalUserContextData": boolean,
   "EnableTokenRevocation": boolean,
   "ExplicitAuthFlows": [ "string" ],
   "IdTokenValidity": number,
   "LogoutURLs": [ "string" ],
   "PreventUserExistenceErrors": "string",
   "ReadAttributes": [ "string" ],
   "RefreshTokenValidity": number,
   "SupportedIdentityProviders": [ "string" ],
   "TokenValidityUnits": {
      "AccessToken": "string",
      "IdToken": "string",
      "RefreshToken": "string"
   },
   "UserPoolId": "string",
   "WriteAttributes": [ "string" ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 450).

The request accepts the following data in JSON format.

AccessTokenValidity (p. 330)

The access token time limit. After this limit expires, your user can't use their access token. To specify the time unit for `AccessTokenValidity` as `seconds`, `minutes`, `hours`, or `days`, set a `TokenValidityUnits` value in your API request.

For example, when you set `AccessTokenValidity` to `10` and `TokenValidityUnits` to `hours`, your user can authorize access with their access token for 10 hours.

The default time unit for `AccessTokenValidity` in an API request is hours. *Valid range* is displayed below in seconds.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 86400.

Required: No

AllowedOAuthFlows (p. 330)

The allowed OAuth flows.

code

Use a code grant flow, which provides an authorization code as the response. This code can be exchanged for access tokens with the `/oauth2/token` endpoint.

implicit

Issue the access token (and, optionally, ID token, based on scopes) directly to your user.

client_credentials

Issue the access token from the `/oauth2/token` endpoint directly to a non-person user using a combination of the client ID and client secret.

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. 330)

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

Type: Boolean

Required: No

AllowedOAuthScopes (p. 330)

The allowed OAuth scopes. Possible values provided by OAuth are `phone`, `email`, `openid`, and `profile`. Possible values provided by AWS are `aws.cognito.signin.user.admin`. Custom scopes created in Resource Servers are also supported.

Type: Array of strings

Array Members: Maximum number of 50 items.

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

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

Required: No

AnalyticsConfiguration (p. 330)

The Amazon Pinpoint analytics configuration necessary to collect metrics for this user pool.

> **Note**
> In AWS Regions where Amazon Pinpoint isn't available, user pools only support sending events to Amazon Pinpoint projects in us-east-1. In Regions where Amazon Pinpoint is available, user pools support sending events to Amazon Pinpoint projects within that same Region.

Type: AnalyticsConfigurationType (p. 356) object

Required: No

CallbackURLs (p. 330)

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

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. 330)

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. 330)

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. 330)

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

EnablePropagateAdditionalUserContextData (p. 330)

Activates the propagation of additional user context data. For more information about propagation of user context data, see Adding advanced security to a user pool. If you don't include this parameter, you can't send device fingerprint information, including source IP address, to Amazon Cognito advanced security. You can only activate `EnablePropagateAdditionalUserContextData` in an app client that has a client secret.

Type: Boolean

Required: No

EnableTokenRevocation (p. 330)

Activates or deactivates token revocation. For more information about revoking tokens, see RevokeToken.

Type: Boolean

Required: No

ExplicitAuthFlows (p. 330)

The authentication flows that are supported by the user pool clients. Flow names without the `ALLOW_` prefix are no longer supported in favor of new names with the `ALLOW_` prefix. Note that values with `ALLOW_` prefix must be used only along with values with the `ALLOW_` prefix.

Valid values include:

- `ALLOW_ADMIN_USER_PASSWORD_AUTH`: Enable admin based user password authentication flow `ADMIN_USER_PASSWORD_AUTH`. This setting replaces the `ADMIN_NO_SRP_AUTH` setting. With this authentication flow, Amazon Cognito receives the password in the request instead of using the Secure Remote Password (SRP) protocol to verify passwords.
- `ALLOW_CUSTOM_AUTH`: Enable Lambda trigger based authentication.
- `ALLOW_USER_PASSWORD_AUTH`: Enable user password-based authentication. In this flow, Amazon Cognito receives the password in the request instead of using the SRP protocol to verify passwords.
- `ALLOW_USER_SRP_AUTH`: Enable SRP-based authentication.
- `ALLOW_REFRESH_TOKEN_AUTH`: Enable authflow to refresh tokens.

Type: Array of strings

Valid Values: `ADMIN_NO_SRP_AUTH | CUSTOM_AUTH_FLOW_ONLY | USER_PASSWORD_AUTH | ALLOW_ADMIN_USER_PASSWORD_AUTH | ALLOW_CUSTOM_AUTH | ALLOW_USER_PASSWORD_AUTH | ALLOW_USER_SRP_AUTH | ALLOW_REFRESH_TOKEN_AUTH`

Required: No

IdTokenValidity (p. 330)

The ID token time limit. After this limit expires, your user can't use their ID token. To specify the time unit for `IdTokenValidity` as `seconds`, `minutes`, `hours`, or `days`, set a `TokenValidityUnits` value in your API request.

For example, when you set `IdTokenValidity` as `10` and `TokenValidityUnits` as `hours`, your user can authenticate their session with their ID token for 10 hours.

The default time unit for `AccessTokenValidity` in an API request is hours. *Valid range* is displayed below in seconds.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 86400.

Required: No

LogoutURLs (p. 330)

A list of allowed logout URLs for the IdPs.

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

PreventUserExistenceErrors (p. 330)

Errors and responses that you want Amazon Cognito APIs to return during authentication, account confirmation, and password recovery when the user doesn't exist in the user pool. When set to `ENABLED` and the user doesn't exist, authentication returns an error indicating either the username or password was incorrect. Account confirmation and password recovery return a response indicating a code was sent to a simulated destination. When set to `LEGACY`, those APIs return a `UserNotFoundException` exception if the user doesn't exist in the user pool.

Valid values include:

- `ENABLED` - This prevents user existence-related errors.
- `LEGACY` - This represents the early behavior of Amazon Cognito where user existence related errors aren't prevented.

This setting affects the behavior of following APIs:

- AdminInitiateAuth (p. 39)
- AdminRespondToAuthChallenge (p. 65)
- InitiateAuth (p. 219)
- RespondToAuthChallenge (p. 261)
- ForgotPassword (p. 186)
- ConfirmForgotPassword (p. 101)
- ConfirmSignUp (p. 106)
- ResendConfirmationCode (p. 256)

Type: String

Valid Values: `LEGACY | ENABLED`

Required: No

ReadAttributes (p. 330)

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. 330)

The refresh token time limit. After this limit expires, your user can't use their refresh token. To specify the time unit for `RefreshTokenValidity` as `seconds`, `minutes`, `hours`, or `days`, set a `TokenValidityUnits` value in your API request.

For example, when you set `RefreshTokenValidity` as `10` and `TokenValidityUnits` as `days`, your user can refresh their session and retrieve new access and ID tokens for 10 days.

The default time unit for `RefreshTokenValidity` in an API request is days. You can't set `RefreshTokenValidity` to 0. If you do, Amazon Cognito overrides the value with the default value of 30 days. *Valid range* is displayed below in seconds.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 315360000.

Required: No

SupportedIdentityProviders (p. 330)

A list of provider names for the IdPs that this client supports. The following are supported: `COGNITO`, `Facebook`, `Google` `LoginWithAmazon`, and the names of your own SAML and OIDC providers.

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

TokenValidityUnits (p. 330)

The units in which the validity times are represented. The default unit for RefreshToken is days, and the default for ID and access tokens is hours.

Type: TokenValidityUnitsType (p. 419) object

Required: No

UserPoolId (p. 330)

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. 330)

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": { 
      "AccessTokenValidity": number,
      "AllowedOAuthFlows": [ "string" ],
      "AllowedOAuthFlowsUserPoolClient": boolean,
      "AllowedOAuthScopes": [ "string" ],
      "AnalyticsConfiguration": { 
         "ApplicationArn": "string",
         "ApplicationId": "string",
         "ExternalId": "string",
         "RoleArn": "string",
         "UserDataShared": boolean
      },
      "CallbackURLs": [ "string" ],
      "ClientId": "string",
      "ClientName": "string",
      "ClientSecret": "string",
      "CreationDate": number,
      "DefaultRedirectURI": "string",
      "EnablePropagateAdditionalUserContextData": boolean,
      "EnableTokenRevocation": boolean,
      "ExplicitAuthFlows": [ "string" ],
      "IdTokenValidity": number,
      "LastModifiedDate": number,
      "LogoutURLs": [ "string" ],
      "PreventUserExistenceErrors": "string",
      "ReadAttributes": [ "string" ],
      "RefreshTokenValidity": number,
      "SupportedIdentityProviders": [ "string" ],
      "TokenValidityUnits": { 
         "AccessToken": "string",
         "IdToken": "string",
         "RefreshToken": "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. 336)

The user pool client value from the response from the server when you request to update the user pool client.

Type: UserPoolClientType (p. 430) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 452).

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 not valid.

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

ScopeDoesNotExistException

This exception is thrown when the specified scope doesn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

UpdateUserPoolDomain

Updates the Secure Sockets Layer (SSL) certificate for the custom domain for your user pool.

You can use this operation to provide the Amazon Resource Name (ARN) of a new certificate to Amazon Cognito. You can't use it to change the domain for a user pool.

A custom domain is used to host the Amazon Cognito hosted UI, which provides sign-up and sign-in pages for your application. When you set up a custom domain, you provide a certificate that you manage with AWS Certificate Manager (ACM). When necessary, you can use this operation to change the certificate that you applied to your custom domain.

Usually, this is unnecessary following routine certificate renewal with ACM. When you renew your existing certificate in ACM, the ARN for your certificate remains the same, and your custom domain uses the new certificate automatically.

However, if you replace your existing certificate with a new one, ACM gives the new certificate a new ARN. To apply the new certificate to your custom domain, you must provide this ARN to Amazon Cognito.

When you add your new certificate in ACM, you must choose US East (N. Virginia) as the AWS Region.

After you submit your request, Amazon Cognito requires up to 1 hour to distribute your new certificate to your custom domain.

For more information about adding a custom domain to your user pool, see Using Your Own Domain for the Hosted UI.

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. 450).

The request accepts the following data in JSON format.

CustomDomainConfig (p. 339)

The configuration for a custom domain that hosts the sign-up and sign-in pages for your application. Use this object to specify an SSL certificate that is managed by ACM.

Type: CustomDomainConfigType (p. 369) object

Required: Yes

Domain (p. 339)

The domain name for the custom domain that hosts the sign-up and sign-in pages for your application. One example might be `auth.example.com`.

This string can include only lowercase letters, numbers, and hyphens. Don't use a hyphen for the first or last character. Use periods to separate subdomain names.

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. 339)

The ID of the user pool that is associated with the custom domain whose certificate you're updating.

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. 340)

The Amazon CloudFront endpoint that Amazon Cognito set up when you added the custom domain to your user pool.

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. 452).

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 isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

VerifySoftwareToken

Use this API to register a user's entered time-based one-time password (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. 450).

The request accepts the following data in JSON format.

AccessToken (p. 342)

A valid access token that Amazon Cognito issued to the user whose software token you want to verify.

Type: String

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

Required: No

FriendlyDeviceName (p. 342)

The friendly device name.

Type: String

Required: No

Session (p. 342)

The session that 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. 342)

The one- time password computed using the secret code returned by AssociateSoftwareToken.

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. 343)

The session that 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. 343)

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. 452).

CodeMismatchException

This exception is thrown if the provided code doesn't 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 not valid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't 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 can't find the requested resource.

HTTP Status Code: 400

SoftwareTokenMFANotFoundException

This exception is thrown when the software token time-based one-time password (TOTP) multi-factor authentication (MFA) isn't activated 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 isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't 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 V2

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

VerifyUserAttribute

Verifies the specified user attributes in the user pool.

If your user pool requires verification before Amazon Cognito updates the attribute value, VerifyUserAttribute updates the affected attribute to its pending value. For more information, see UserAttributeUpdateSettingsType.

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. 450).

The request accepts the following data in JSON format.

AccessToken (p. 346)

A valid access token that Amazon Cognito issued to the user whose user attributes you want to verify.

Type: String

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

Required: Yes

AttributeName (p. 346)

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. 346)

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. 452).

AliasExistsException

This exception is thrown when a user tries to confirm the account with an email address or phone number that has already been supplied as an alias for a different user profile. This exception indicates that an account with this email address or phone already exists in a user pool that you've configured to use email address or phone number as a sign-in alias.

HTTP Status Code: 400

CodeMismatchException

This exception is thrown if the provided code doesn't 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 isn't 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 can't 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 isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't 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 V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

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:

- AccountRecoverySettingType (p. 351)
- AccountTakeoverActionsType (p. 352)
- AccountTakeoverActionType (p. 353)
- AccountTakeoverRiskConfigurationType (p. 354)
- AdminCreateUserConfigType (p. 355)
- AnalyticsConfigurationType (p. 356)
- AnalyticsMetadataType (p. 358)
- AttributeType (p. 359)
- AuthenticationResultType (p. 360)
- AuthEventType (p. 362)
- ChallengeResponseType (p. 364)
- CodeDeliveryDetailsType (p. 365)
- CompromisedCredentialsActionsType (p. 366)
- CompromisedCredentialsRiskConfigurationType (p. 367)
- ContextDataType (p. 368)
- CustomDomainConfigType (p. 369)
- CustomEmailLambdaVersionConfigType (p. 370)
- CustomSMSLambdaVersionConfigType (p. 371)
- DeviceConfigurationType (p. 372)
- DeviceSecretVerifierConfigType (p. 373)
- DeviceType (p. 374)
- DomainDescriptionType (p. 375)
- EmailConfigurationType (p. 377)
- EventContextDataType (p. 380)
- EventFeedbackType (p. 381)
- EventRiskType (p. 382)
- GroupType (p. 383)
- HttpHeader (p. 385)
- IdentityProviderType (p. 386)
- LambdaConfigType (p. 389)
- MessageTemplateType (p. 392)
- MFAOptionType (p. 393)
- NewDeviceMetadataType (p. 394)
- NotifyConfigurationType (p. 395)
- NotifyEmailType (p. 397)

AccountRecoverySettingType

The data type for `AccountRecoverySetting`.

Contents

RecoveryMechanisms

The list of `RecoveryOptionTypes`.

Type: Array of RecoveryOptionType (p. 403) objects

Array Members: Minimum number of 1 item. Maximum number of 2 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 V2
- AWS SDK for Ruby V3

AccountTakeoverActionsType

Account takeover actions type.

Contents

HighAction

Action to take for a high risk.

Type: AccountTakeoverActionType (p. 353) object

Required: No

LowAction

Action to take for a low risk.

Type: AccountTakeoverActionType (p. 353) object

Required: No

MediumAction

Action to take for a medium risk.

Type: AccountTakeoverActionType (p. 353) 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 V2
- AWS SDK for Ruby V3

AccountTakeoverActionType

Account takeover action type.

Contents

EventAction

The action to take in response to the account takeover action. Valid values are as follows:

- `BLOCK` Choosing this action will block the request.
- `MFA_IF_CONFIGURED` Present an MFA challenge if user has configured it, else allow the request.
- `MFA_REQUIRED` Present an MFA challenge if user has configured it, else block the request.
- `NO_ACTION` Allow the user to 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 V2
- AWS SDK for Ruby V3

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. 352) object

Required: Yes

NotifyConfiguration

The notify configuration used to construct email notifications.

Type: NotifyConfigurationType (p. 395) 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 V2
- AWS SDK for Ruby V3

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. 392) object

Required: No

UnusedAccountValidityDays

The user account expiration limit, in days, after which a new account that hasn't signed in 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.

> **Note**
> If you set a value for `TemporaryPasswordValidityDays` in `PasswordPolicy`, that value will be used, and `UnusedAccountValidityDays` will be no longer be an available parameter for that user pool.

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 V2
- AWS SDK for Ruby V3

AnalyticsConfigurationType

The Amazon Pinpoint analytics configuration necessary to collect metrics for a user pool.

> **Note**
> In Regions where Amazon Pinpointisn't available, user pools only support sending events to Amazon Pinpoint projects in us-east-1. In Regions where Amazon Pinpoint is available, user pools support sending events to Amazon Pinpoint projects within that same Region.

Contents

ApplicationArn

The Amazon Resource Name (ARN) of an Amazon Pinpoint project. You can use the Amazon Pinpoint project to integrate with the chosen user pool Client. Amazon Cognito publishes events to the Amazon Pinpoint project that the app ARN declares.

Type: String

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

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

Required: No

ApplicationId

The application ID for an Amazon Pinpoint application.

Type: String

Pattern: `^[0-9a-fA-F]+$`

Required: No

ExternalId

The external ID.

Type: String

Required: No

RoleArn

The ARN of an AWS Identity and Access Management 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: No

UserDataShared

If `UserDataShared` is `true`, Amazon Cognito includes user data in the events that 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 V2
- AWS SDK for Ruby V3

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. For more information about AWS Regions that can contain Amazon Pinpoint resources for use with Amazon Cognito user pools, see Using Amazon Pinpoint analytics with Amazon Cognito user pools.

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 V2
- AWS SDK for Ruby V3

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 V2
- AWS SDK for Ruby V3

AuthenticationResultType

The authentication result.

Contents

AccessToken

A valid access token that Amazon Cognito issued to the user who you want to authenticate.

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. 394) 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 V2
- AWS SDK for Ruby V3

AuthEventType

The authentication event type.

Contents

ChallengeResponses

The challenge responses.

Type: Array of ChallengeResponseType (p. 364) objects

Required: No

CreationDate

The creation date

Type: Timestamp

Required: No

EventContextData

The user context data captured at the time of an event request. This value provides additional information about the client from which event the request is received.

Type: EventContextDataType (p. 380) 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. 381) 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. 382) 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 V2
- AWS SDK for Ruby V3

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 V2
- AWS SDK for Ruby V3

CodeDeliveryDetailsType

The delivery details for an email or SMS message that Amazon Cognito sent for authentication or verification.

Contents

AttributeName

The name of the attribute that Amazon Cognito verifies with the 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: No

DeliveryMedium

The method that Amazon Cognito used to send the code.

Type: String

Valid Values: `SMS | EMAIL`

Required: No

Destination

The email address or phone number destination where Amazon Cognito sent the code.

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 V2
- AWS SDK for Ruby V3

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 V2
- AWS SDK for Ruby V3

CompromisedCredentialsRiskConfigurationType

The compromised credentials risk configuration type.

Contents

Actions

The compromised credentials risk configuration actions.

Type: CompromisedCredentialsActionsType (p. 366) 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 V2
- AWS SDK for Ruby V3

ContextDataType

Contextual user data type used for evaluating the risk of an unexpected event by Amazon Cognito advanced security.

Contents

EncodedData

Encoded device-fingerprint details that your app collected with the Amazon Cognito context data collection library. For more information, see Adding user device and session data to API requests.

Type: String

Required: No

HttpHeaders

HttpHeaders received on your server in same order.

Type: Array of HttpHeader (p. 385) objects

Required: Yes

IpAddress

The source IP address of your user's device.

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 V2
- AWS SDK for Ruby V3

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 V2
- AWS SDK for Ruby V3

CustomEmailLambdaVersionConfigType

A custom email sender AWS Lambda configuration type.

Contents

LambdaArn

The Amazon Resource Name (ARN) of the Lambda function that Amazon Cognito activates to send email notifications to users.

Type: String

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

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

Required: Yes

LambdaVersion

Signature of the "request" attribute in the "event" information Amazon Cognito passes to your custom email Lambda function. The only supported value is `V1_0`.

Type: String

Valid Values: `V1_0`

Required: Yes

See Also

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

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3

CustomSMSLambdaVersionConfigType

A custom SMS sender AWS Lambda configuration type.

Contents

LambdaArn

The Amazon Resource Name (ARN) of the Lambda function that Amazon Cognito activates to send SMS notifications to users.

Type: String

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

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

Required: Yes

LambdaVersion

Signature of the "request" attribute in the "event" information that Amazon Cognito passes to your custom SMS Lambda function. The only supported value is `V1_0`.

Type: String

Valid Values: `V1_0`

Required: Yes

See Also

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

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3

DeviceConfigurationType

The device tracking configuration for a user pool. A user pool with device tracking deactivated returns a null value.

> **Note**
> When you provide values for any DeviceConfiguration field, you activate device tracking.

Contents

ChallengeRequiredOnNewDevice

When true, device authentication can replace SMS and time-based one-time password (TOTP) factors for multi-factor authentication (MFA).

> **Note**
> Users that sign in with devices that have not been confirmed or remembered will still have to provide a second factor, whether or not ChallengeRequiredOnNewDevice is true, when your user pool requires MFA.

Type: Boolean

Required: No

DeviceOnlyRememberedOnUserPrompt

When true, users can opt in to remembering their device. Your app code must use callback functions to return the user's choice.

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 V2
- AWS SDK for Ruby V3

DeviceSecretVerifierConfigType

The device verifier against which it is 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 V2
- AWS SDK for Ruby V3

DeviceType

The device type.

Contents

DeviceAttributes

The device attributes.

Type: Array of AttributeType (p. 359) 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 when 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 V2
- AWS SDK for Ruby V3

DomainDescriptionType

A container for information about a domain.

Contents

AWSAccountId

The AWS ID for the user pool owner.

Type: String

Required: No

CloudFrontDistribution

The Amazon Resource Name (ARN) of the Amazon 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. 369) object

Required: No

Domain

The domain string. For custom domains, this is the fully-qualified domain name, such as `auth.example.com`. For Amazon Cognito prefix domains, this is the prefix alone, such as `auth`.

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 Amazon 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 V2
- AWS SDK for Ruby V3

EmailConfigurationType

The email configuration of your user pool. The email configuration type sets your preferred sending method, AWS Region, and sender for messages from your user pool.

> **Note**
> Amazon Cognito can send email messages with Amazon Simple Email Service resources in the AWS Region where you created your user pool, and in alternate Regions in some cases. For more information on the supported Regions, see Email settings for Amazon Cognito user pools.

Contents

ConfigurationSet

The set of configuration rules that can be applied to emails sent using Amazon Simple Email Service. A configuration set is applied to an email by including a reference to the configuration set in the headers of the email. Once applied, all of the rules in that configuration set are applied to the email. Configuration sets can be used to apply the following types of rules to emails:

Event publishing

Amazon Simple Email Service can track the number of send, delivery, open, click, bounce, and complaint events for each email sent. Use event publishing to send information about these events to other AWS services such as and Amazon CloudWatch

IP pool management

When leasing dedicated IP addresses with Amazon Simple Email Service, you can create groups of IP addresses, called dedicated IP pools. You can then associate the dedicated IP pools with configuration sets.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `^[a-zA-Z0-9_-]+$`

Required: No

EmailSendingAccount

Specifies whether Amazon Cognito uses its built-in functionality to send your users email messages, or uses your Amazon Simple Email Service email configuration. Specify one of the following values:

COGNITO_DEFAULT

When Amazon Cognito emails your users, it uses its built-in email functionality. When you use the default option, Amazon Cognito allows only a limited number of emails each day for your user pool. For typical production environments, the default email limit is less than the required delivery volume. To achieve a higher delivery volume, specify DEVELOPER to use your Amazon SES email configuration.

To look up the email delivery limit for the default option, see Limits in in the *Developer Guide*.

The default FROM address is `no-reply@verificationemail.com`. To customize the FROM address, provide the Amazon Resource Name (ARN) of an Amazon SES verified email address for the `SourceArn` parameter.

DEVELOPER

When Amazon Cognito emails your users, it uses your Amazon SES configuration. Amazon Cognito calls Amazon SES on your behalf to send email from your verified email address. When you use this option, the email delivery limits are the same limits that apply to your Amazon SES verified email address in your AWS account.

If you use this option, provide the ARN of an Amazon SES verified email address for the `SourceArn` parameter.

Before Amazon Cognito can email your users, it requires additional permissions to call Amazon SES on your behalf. When you update your user pool with this option, Amazon Cognito creates a *service-linked role*, which is a type of role, in your AWS account. This role contains the permissions that allow to access Amazon SES and send email messages with your address. For more information about the service-linked role that Amazon Cognito creates, see Using Service-Linked Roles for Amazon Cognito in the *Amazon Cognito Developer Guide*.

Type: String

Valid Values: `COGNITO_DEFAULT | DEVELOPER`

Required: No

From

Either the sender’s email address or the sender’s name with their email address. For example, `testuser@example.com` or `Test User <testuser@example.com>`. This address appears before the body of the email.

Type: String

Required: No

ReplyToEmailAddress

The destination to which the receiver of the email should reply.

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 ARN of a verified email address in Amazon SES. Amazon Cognito uses this email address in one of the following ways, depending on the value that you specify for the `EmailSendingAccount` parameter:

- If you specify `COGNITO_DEFAULT`, Amazon Cognito uses this address as the custom FROM address when it emails your users using its built-in email account.
- If you specify `DEVELOPER`, Amazon Cognito emails your users with this address by calling Amazon SES on your behalf.

The Region value of the `SourceArn` parameter must indicate a supported AWS Region of your user pool. Typically, the Region in the `SourceArn` and the user pool Region are the same. For more information, see Amazon SES email configuration regions 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

See Also

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

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3

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 source IP address of your user's device.

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 V2
- AWS SDK for Ruby V3

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 V2
- AWS SDK for Ruby V3

EventRiskType

The event risk type.

Contents

CompromisedCredentialsDetected

Indicates whether compromised credentials were detected during an authentication event.

Type: Boolean

Required: No

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 V2
- AWS SDK for Ruby V3

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 non-negative 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 ornull `Precedence` values. If a user belongs to two or more groups, it is the group with the lowest precedence value whose role ARN is given in the user's tokens for the `cognito:roles` and `cognito:preferred_role` claims.

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 isn't set in users' tokens.

The default `Precedence` value is null.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

RoleArn

The role Amazon Resource Name (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 V2
- AWS SDK for Ruby V3

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 V2
- AWS SDK for Ruby V3

IdentityProviderType

A container for information about an IdP.

Contents

AttributeMapping

A mapping of IdP 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 IdP was created.

Type: Timestamp

Required: No

IdpIdentifiers

A list of IdP 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 IdP was last modified.

Type: Timestamp

Required: No

ProviderDetails

The IdP details. The following list describes the provider detail keys for each IdP type.

- For Google and Login with Amazon:
  - client_id
  - client_secret
  - authorize_scopes
- For Facebook:
  - client_id
  - client_secret
  - authorize_scopes
  - api_version
- For Sign in with Apple:

    - client_id
    - team_id
    - key_id
    - private_key

      You can submit a private_key when you add or update an IdP. Describe operations don't return the private key.
    - authorize_scopes
  - For OIDC providers:
    - client_id
    - client_secret
    - attributes_request_method
    - oidc_issuer
    - authorize_scopes
    - The following keys are only present if Amazon Cognito didn't discover them at the `oidc_issuer` URL.
      - authorize_url
      - token_url
      - attributes_url
      - jwks_uri
    - Amazon Cognito sets the value of the following keys automatically. They are read-only.
      - attributes_url_add_attributes
  - For SAML providers:
    - MetadataFile or MetadataURL
    - IDPSignout *optional*

  Type: String to string map

  Required: No

ProviderName

The IdP 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 IdP type.

Type: String

Valid Values: `SAML | Facebook | Google | LoginWithAmazon | SignInWithApple | 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 V2
- AWS SDK for Ruby V3

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

CustomEmailSender

A custom email sender Lambda trigger.

Type: CustomEmailLambdaVersionConfigType (p. 370) object

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

CustomSMSSender

A custom SMS sender Lambda trigger.

Type: CustomSMSLambdaVersionConfigType (p. 371) object

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

KMSKeyID

The Amazon Resource Name (ARN) of an AWS KMS key. Amazon Cognito uses the key to encrypt codes and temporary passwords sent to `CustomEmailSender` and `CustomSMSSender`.

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 V2
- AWS SDK for Ruby V3

MessageTemplateType

The message template structure.

Contents

EmailMessage

The message template for email messages. EmailMessage is allowed only if EmailSendingAccount is DEVELOPER.

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. EmailSubject is allowed only if EmailSendingAccount is DEVELOPER.

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 V2
- AWS SDK for Ruby V3

MFAOptionType

This data type is no longer supported. Applies only to SMS multi-factor authentication (MFA) configurations. Does not apply to time-based one-time password (TOTP) software token MFA configurations.

To set either type of MFA configuration, use the AdminSetUserMFAPreference (p. 72) or SetUserMFAPreference (p. 278) actions.

To look up information about either type of MFA configuration, use the AdminGetUser:UserMFASettingList (p. 36) or GetUser:UserMFASettingList (p. 208) responses.

Contents

AttributeName

The attribute name of the MFA option type. The only valid value is `phone_number`.

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 to send the MFA code. You can use this parameter to set only the `SMS` delivery medium value.

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 V2
- AWS SDK for Ruby V3

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 V2
- AWS SDK for Ruby V3

NotifyConfigurationType

The notify configuration type.

Contents

BlockEmail

Email template used when a detected risk event is blocked.

Type: NotifyEmailType (p. 397) object

Required: No

From

The email address that is sending the email. The address must be either individually verified with Amazon Simple Email Service, or from a domain that has been verified with Amazon SES.

Type: String

Required: No

MfaEmail

The multi-factor authentication (MFA) email template used when MFA is challenged as part of a detected risk.

Type: NotifyEmailType (p. 397) object

Required: No

NoActionEmail

The email template used when a detected risk event is allowed.

Type: NotifyEmailType (p. 397) 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. This identity 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 V2
- AWS SDK for Ruby V3

NotifyEmailType

The notify email type.

Contents

HtmlBody

The email 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 email 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 email 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 V2
- AWS SDK for Ruby V3

NumberAttributeConstraintsType

The minimum and maximum values 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 V2
- AWS SDK for Ruby V3

PasswordPolicyType

The password policy type.

Contents

MinimumLength

The minimum length of the password in the policy that you have set. This value can't 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

TemporaryPasswordValidityDays

The number of days a temporary password is valid in the password policy. If the user doesn't sign in during this time, an administrator must reset their password.

> **Note**
> When you set `TemporaryPasswordValidityDays` for a user pool, you can no longer set a value for the legacy `UnusedAccountValidityDays` parameter in that user pool.

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 V2
- AWS SDK for Ruby V3

ProviderDescription

A container for IdP 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 IdP 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 IdP type.

Type: String

Valid Values: `SAML | Facebook | Google | LoginWithAmazon | SignInWithApple | 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 V2
- AWS SDK for Ruby V3

ProviderUserIdentifierType

A container for information about an IdP for a user pool.

Contents

ProviderAttributeName

The name of the provider attribute to link to, such as `NameID`.

Type: String

Required: No

ProviderAttributeValue

The value of the provider attribute to link to, such as `xxxxx_account`.

Type: String

Required: No

ProviderName

The name of the provider, such as 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 V2
- AWS SDK for Ruby V3

RecoveryOptionType

A map containing a priority as a key, and recovery method name as a value.

Contents

Name

The recovery method for a user.

Type: String

Valid Values: `verified_email | verified_phone_number | admin_only`

Required: Yes

Priority

A positive integer specifying priority of a method with 1 being the highest priority.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 2.

Required: Yes

See Also

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

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3

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 V2
- AWS SDK for Ruby V3

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. 404) objects

Array Members: Maximum number of 100 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 V2

- AWS SDK for Ruby V3

RiskConfigurationType

The risk configuration type.

Contents

AccountTakeoverRiskConfiguration

The account takeover risk configuration object, including the `NotifyConfiguration` object and `Actions` to take if there is an account takeover.

Type: AccountTakeoverRiskConfigurationType (p. 354) 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. 367) object

Required: No

LastModifiedDate

The last modified date.

Type: Timestamp

Required: No

RiskExceptionConfiguration

The configuration to override the risk decision.

Type: RiskExceptionConfigurationType (p. 409) 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 V2
- AWS SDK for Ruby V3

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 routing prefix.

Type: Array of strings

Array Members: Maximum number of 200 items.

Required: No

SkippedIPRangeList

Risk detection isn't performed on the IP addresses in this range list. The IP range is in CIDR notation.

Type: Array of strings

Array Members: Maximum number of 200 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 V2
- AWS SDK for Ruby V3

SchemaAttributeType

Contains information about the schema attribute.

Contents

AttributeDataType

The attribute data type.

Type: String

Valid Values: `String | Number | DateTime | Boolean`

Required: No

DeveloperOnlyAttribute

> **Note**
> You should use WriteAttributes in the user pool client to control how attributes can be mutated for new use cases instead of using `DeveloperOnlyAttribute`.

Specifies whether the attribute type is developer only. This attribute can only be modified by an administrator. Users won't be able to modify this attribute using their access token. For example, `DeveloperOnlyAttribute` can be modified using AdminUpdateUserAttributes but can't be updated using UpdateUserAttributes.

Type: Boolean

Required: No

Mutable

Specifies whether the value of the attribute can be changed.

For any user pool attribute that is mapped to an IdP attribute, you must set this parameter to `true`. Amazon Cognito updates mapped attributes when users sign in to your application through an IdP. 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. 398) object

Required: No

Required

Specifies whether a user pool attribute is required. If the attribute is required and the user doesn't 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. 418) 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 V2
- AWS SDK for Ruby V3

SmsConfigurationType

The SMS configuration type is the settings that your Amazon Cognito user pool must use to send an SMS message from your AWS account through Amazon Simple Notification Service. To send SMS messages with Amazon SNS in the AWS Region that you want, the Amazon Cognito user pool uses an AWS Identity and Access Management (IAM) role in your AWS account.

Contents

ExternalId

The external ID provides additional security for your IAM role. You can use an `ExternalId` with the IAM role that you use with Amazon SNS to send SMS messages for your user pool. If you provide an `ExternalId`, your Amazon Cognito user pool includes it in the request to assume your IAM role. You can configure the role trust policy to require that Amazon Cognito, and any principal, provide the `ExternalID`. If you use the Amazon Cognito Management Console to create a role for SMS multi-factor authentication (MFA), Amazon Cognito creates a role with the required permissions and a trust policy that demonstrates use of the `ExternalId`.

For more information about the `ExternalId` of a role, see How to use an external ID when granting access to your AWS resources to a third party

Type: String

Required: No

SnsCallerArn

The Amazon Resource Name (ARN) of the Amazon SNS caller. This is the ARN of the IAM role in your AWS account that Amazon Cognito will use to send SMS messages. SMS messages are subject to a spending limit.

Type: String

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

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

Required: Yes

SnsRegion

The AWS Region to use with Amazon SNS integration. You can choose the same Region as your user pool, or a supported **Legacy Amazon SNS alternate Region**.

Amazon Cognito resources in the Asia Pacific (Seoul) AWS Region must use your Amazon SNS configuration in the Asia Pacific (Tokyo) Region. For more information, see SMS message settings for Amazon Cognito user pools.

Type: String

Length Constraints: Minimum length of 5. Maximum length of 32.

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 V2
- AWS SDK for Ruby V3

SmsMfaConfigType

The SMS text message multi-factor authentication (MFA) configuration type.

Contents

SmsAuthenticationMessage

The SMS authentication message that will be sent to users with the code they must sign in. The message must contain the ‘{####}’ placeholder, which is replaced with the code. If the message isn't included, and default message will be used.

Type: String

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

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

Required: No

SmsConfiguration

The SMS configuration with the settings that your Amazon Cognito user pool must use to send an SMS message from your AWS account through Amazon Simple Notification Service. To request Amazon SNS in the AWS Region that you want, the Amazon Cognito user pool uses an AWS Identity and Access Management (IAM) role that you provide for your AWS account.

Type: SmsConfigurationType (p. 412) 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 V2
- AWS SDK for Ruby V3

SMSMfaSettingsType

The type used for enabling SMS multi-factor authentication (MFA) at the user level. Phone numbers don't need to be verified to be used for SMS MFA. If an MFA type is activated for a user, the user will be prompted for MFA during all sign-in attempts, unless device tracking is turned on and the device has been trusted. If you would like MFA to be applied selectively based on the assessed risk level of sign-in attempts, deactivate MFA for users and turn on Adaptive Authentication for the user pool.

Contents

Enabled

Specifies whether SMS text message MFA is activated. If an MFA type is activated for a user, the user will be prompted for MFA during all sign-in attempts, unless device tracking is turned on and the device has been trusted.

Type: Boolean

Required: No

PreferredMfa

Specifies whether SMS is 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 V2
- AWS SDK for Ruby V3

SoftwareTokenMfaConfigType

The type used for enabling software token MFA at the user pool level.

Contents

Enabled

Specifies whether software token MFA is activated.

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 V2
- AWS SDK for Ruby V3

SoftwareTokenMfaSettingsType

The type used for enabling software token MFA at the user level. If an MFA type is activated for a user, the user will be prompted for MFA during all sign-in attempts, unless device tracking is turned on and the device has been trusted. If you want MFA to be applied selectively based on the assessed risk level of sign-in attempts, deactivate MFA for users and turn on Adaptive Authentication for the user pool.

Contents

Enabled

Specifies whether software token MFA is activated. If an MFA type is activated for a user, the user will be prompted for MFA during all sign-in attempts, unless device tracking is turned on and the device has been trusted.

Type: Boolean

Required: No

PreferredMfa

Specifies whether software token MFA is 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 V2
- AWS SDK for Ruby V3

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 V2
- AWS SDK for Ruby V3

TokenValidityUnitsType

The data type TokenValidityUnits specifies the time units you use when you set the duration of ID, access, and refresh tokens.

Contents

AccessToken

A time unit of `seconds`, `minutes`, `hours`, or `days` for the value that you set in the `AccessTokenValidity` parameter. The default `AccessTokenValidity` time unit is hours.

Type: String

Valid Values: `seconds | minutes | hours | days`

Required: No

IdToken

A time unit of `seconds`, `minutes`, `hours`, or `days` for the value that you set in the `IdTokenValidity` parameter. The default `IdTokenValidity` time unit is hours.

Type: String

Valid Values: `seconds | minutes | hours | days`

Required: No

RefreshToken

A time unit of `seconds`, `minutes`, `hours`, or `days` for the value that you set in the `RefreshTokenValidity` parameter. The default `RefreshTokenValidity` time unit is days.

Type: String

Valid Values: `seconds | minutes | hours | days`

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 V2
- AWS SDK for Ruby V3

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 V2
- AWS SDK for Ruby V3

UserAttributeUpdateSettingsType

The settings for updates to user attributes. These settings include the property `AttributesRequireVerificationBeforeUpdate`, a user-pool setting that tells Amazon Cognito how to handle changes to the value of your users' email address and phone number attributes. For more information, see Verifying updates to to email addresses and phone numbers.

Contents

AttributesRequireVerificationBeforeUpdate

Requires that your user verifies their email address, phone number, or both before Amazon Cognito updates the value of that attribute. When you update a user attribute that has this option activated, Amazon Cognito sends a verification message to the new phone number or email address. Amazon Cognito doesn’t change the value of the attribute until your user responds to the verification message and confirms the new value.

You can verify an updated email address or phone number with a VerifyUserAttribute API request. You can also call the UpdateUserAttributes or AdminUpdateUserAttributes API and set `email_verified` or `phone_number_verified` to true.

When `AttributesRequireVerificationBeforeUpdate` is false, your user pool doesn't require that your users verify attribute changes before Amazon Cognito updates them. In a user pool where `AttributesRequireVerificationBeforeUpdate` is false, API operations that change attribute values can immediately update a user’s `email` or `phone_number` attribute.

Type: Array of strings

Valid Values: `phone_number | 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 V2
- AWS SDK for Ruby V3

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

Encoded device-fingerprint details that your app collected with the Amazon Cognito context data collection library. For more information, see Adding user device and session data to API requests.

Type: String

Required: No

IpAddress

The source IP address of your user's device.

Type: String

Required: No

See Also

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

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3

UserImportJobType

The user import job type.

Contents

CloudWatchLogsRoleArn

The role Amazon Resource Name (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 couldn't 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 can't 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 V2
- AWS SDK for Ruby V3

UsernameConfigurationType

The username configuration type.

Contents

CaseSensitive

Specifies whether user name case sensitivity will be applied for all users in the user pool through Amazon Cognito APIs.

Valid values include:

True

Enables case sensitivity for all username input. When this option is set to `True`, users must sign in using the exact capitalization of their given username, such as “UserName”. This is the default value.

False

Enables case insensitivity for all username input. For example, when this option is set to `False`, users can sign in using either "username" or "Username". This option also enables both `preferred_username` and `email` alias to be case insensitive, in addition to the `username` attribute.

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 V2
- AWS SDK for Ruby V3

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 V2
- AWS SDK for Ruby V3

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 V2
- AWS SDK for Ruby V3

UserPoolClientType

Contains information about a user pool client.

Contents

AccessTokenValidity

The access token time limit. After this limit expires, your user can't use their access token. To specify the time unit for `AccessTokenValidity` as `seconds`, `minutes`, `hours`, or `days`, set a `TokenValidityUnits` value in your API request.

For example, when you set `AccessTokenValidity` to `10` and `TokenValidityUnits` to `hours`, your user can authorize access with their access token for 10 hours.

The default time unit for `AccessTokenValidity` in an API request is hours. *Valid range* is displayed below in seconds.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 86400.

Required: No

AllowedOAuthFlows

The allowed OAuth flows.

code

Use a code grant flow, which provides an authorization code as the response. This code can be exchanged for access tokens with the `/oauth2/token` endpoint.

implicit

Issue the access token (and, optionally, ID token, based on scopes) directly to your user.

client_credentials

Issue the access token from the `/oauth2/token` endpoint directly to a non-person user using a combination of the client ID and client secret.

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 Amazon Cognito user pools.

Type: Boolean

Required: No

AllowedOAuthScopes

The OAuth scopes that your app client supports. Possible values that OAuth provides are `phone`, `email`, `openid`, and `profile`. Possible values that AWS provides are `aws.cognito.signin.user.admin`. Amazon Cognito also supports custom scopes that you create in Resource Servers.

Type: Array of strings

Array Members: Maximum number of 50 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.

> **Note**
> Amazon Cognito user pools only support sending events to Amazon Pinpoint projects in the US East (N. Virginia) us-east-1 Region, regardless of the Region where the user pool resides.

Type: AnalyticsConfigurationType (p. 356) object

Required: No

CallbackURLs

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

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

EnablePropagateAdditionalUserContextData

When `EnablePropagateAdditionalUserContextData` is true, Amazon Cognito accepts an `IpAddress` value that you send in the `UserContextData` parameter. The `UserContextData` parameter sends information to Amazon Cognito advanced security for risk analysis. You can send `UserContextData` when you sign in Amazon Cognito native users with the `InitiateAuth` and `RespondToAuthChallenge` API operations.

When `EnablePropagateAdditionalUserContextData` is false, you can't send your user's source IP address to Amazon Cognito advanced security with unauthenticated API operations. `EnablePropagateAdditionalUserContextData` doesn't affect whether you can send a source IP address in a `ContextData` parameter with the authenticated API operations `AdminInitiateAuth` and `AdminRespondToAuthChallenge`.

You can only activate `EnablePropagateAdditionalUserContextData` in an app client that has a client secret. For more information about propagation of user context data, see Adding user device and session data to API requests.

Type: Boolean

Required: No

EnableTokenRevocation

Indicates whether token revocation is activated for the user pool client. When you create a new user pool client, token revocation is activated by default. For more information about revoking tokens, see RevokeToken.

Type: Boolean

Required: No

ExplicitAuthFlows

The authentication flows that are supported by the user pool clients. Flow names without the `ALLOW_` prefix are no longer supported in favor of new names with the `ALLOW_` prefix. Note that values with `ALLOW_` prefix must be used only along with values including the `ALLOW_` prefix.

Valid values include:

- `ALLOW_ADMIN_USER_PASSWORD_AUTH`: Enable admin based user password authentication flow `ADMIN_USER_PASSWORD_AUTH`. This setting replaces the `ADMIN_NO_SRP_AUTH` setting. With this authentication flow, Amazon Cognito receives the password in the request instead of using the Secure Remote Password (SRP) protocol to verify passwords.
- `ALLOW_CUSTOM_AUTH`: Enable Lambda trigger based authentication.
- `ALLOW_USER_PASSWORD_AUTH`: Enable user password-based authentication. In this flow, Amazon Cognito receives the password in the request instead of using the SRP protocol to verify passwords.
- `ALLOW_USER_SRP_AUTH`: Enable SRP-based authentication.
- `ALLOW_REFRESH_TOKEN_AUTH`: Enable authflow to refresh tokens.

Type: Array of strings

Valid Values: `ADMIN_NO_SRP_AUTH | CUSTOM_AUTH_FLOW_ONLY | USER_PASSWORD_AUTH | ALLOW_ADMIN_USER_PASSWORD_AUTH | ALLOW_CUSTOM_AUTH | ALLOW_USER_PASSWORD_AUTH | ALLOW_USER_SRP_AUTH | ALLOW_REFRESH_TOKEN_AUTH`

Required: No

IdTokenValidity

The ID token time limit. After this limit expires, your user can't use their ID token. To specify the time unit for `IdTokenValidity` as `seconds`, `minutes`, `hours`, or `days`, set a `TokenValidityUnits` value in your API request.

For example, when you set `IdTokenValidity` as `10` and `TokenValidityUnits` as `hours`, your user can authenticate their session with their ID token for 10 hours.

The default time unit for `AccessTokenValidity` in an API request is hours. *Valid range* is displayed below in seconds.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 86400.

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 IdPs.

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

PreventUserExistenceErrors

Errors and responses that you want Amazon Cognito APIs to return during authentication, account confirmation, and password recovery when the user doesn't exist in the user pool. When set to `ENABLED` and the user doesn't exist, authentication returns an error indicating either the username or password was incorrect. Account confirmation and password recovery return a response indicating a code was sent to a simulated destination. When set to `LEGACY`, those APIs return a `UserNotFoundException` exception if the user doesn't exist in the user pool.

Valid values include:

- `ENABLED` - This prevents user existence-related errors.
- `LEGACY` - This represents the old behavior of Amazon Cognito where user existence related errors aren't prevented.

This setting affects the behavior of following APIs:

- AdminInitiateAuth (p. 39)
- AdminRespondToAuthChallenge (p. 65)
- InitiateAuth (p. 219)
- RespondToAuthChallenge (p. 261)
- ForgotPassword (p. 186)
- ConfirmForgotPassword (p. 101)
- ConfirmSignUp (p. 106)
- ResendConfirmationCode (p. 256)

Type: String

Valid Values: `LEGACY | ENABLED`

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 refresh token time limit. After this limit expires, your user can't use their refresh token. To specify the time unit for `RefreshTokenValidity` as `seconds`, `minutes`, `hours`, or `days`, set a `TokenValidityUnits` value in your API request.

For example, when you set `RefreshTokenValidity` as `10` and `TokenValidityUnits` as `days`, your user can refresh their session and retrieve new access and ID tokens for 10 days.

The default time unit for `RefreshTokenValidity` in an API request is days. You can't set `RefreshTokenValidity` to 0. If you do, Amazon Cognito overrides the value with the default value of 30 days. *Valid range* is displayed below in seconds.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 315360000.

Required: No

SupportedIdentityProviders

A list of provider names for the IdPs that this client supports. The following are supported: `COGNITO`, `Facebook`, `Google` `LoginWithAmazon`, and the names of your own SAML and OIDC providers.

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

TokenValidityUnits

The time units used to specify the token validity times of each token type: ID, access, and refresh.

Type: TokenValidityUnitsType (p. 419) object

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 V2
- AWS SDK for Ruby V3

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. 389) 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 V2
- AWS SDK for Ruby V3

UserPoolPolicyType

The policy associated with a user pool.

Contents

PasswordPolicy

The password policy.

Type: PasswordPolicyType (p. 399) 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 V2
- AWS SDK for Ruby V3

UserPoolType

A container for information about the user pool.

Contents

AccountRecoverySetting

The available verified method a user can use to recover their password when they call `ForgotPassword`. You can use this setting to define a preferred method when a user has more than one method available. With this setting, SMS doesn't qualify for a valid password recovery mechanism if the user also has SMS multi-factor authentication (MFA) activated. In the absence of this setting, Amazon Cognito uses the legacy behavior to determine the recovery method where SMS is preferred through email.

Type: AccountRecoverySettingType (p. 351) object

Required: No

AdminCreateUserConfig

The configuration for `AdminCreateUser` requests.

Type: AdminCreateUserConfigType (p. 355) object

Required: No

AliasAttributes

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

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

A custom domain name that you provide to Amazon Cognito. This parameter applies only if you use a custom domain to host the sign-up and sign-in pages for your application. An example of a custom domain name might be `auth.example.com`.

For more information about adding a custom domain to your user pool, see Using Your Own Domain for the Hosted UI.

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. 372) object

Required: No

Domain

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 of your user pool. The email configuration type sets your preferred sending method, AWS Region, and sender for messages tfrom your user pool.

Type: EmailConfigurationType (p. 377) object

Required: No

EmailConfigurationFailure

Deprecated. Review error codes from API requests with `EventSource:cognito-idp.amazonaws.com` in AWS CloudTrail for information about problems with user pool email configuration.

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. 389) 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 aren't required and can't be specified during user registration.
- `ON` - MFA tokens are required for all user registrations. You can only specify required when you're 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. 439) object

Required: No

SchemaAttributes

A container with the schema attributes of a user pool.

Type: Array of SchemaAttributeType (p. 410) 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 with the settings that your Amazon Cognito user pool must use to send an SMS message from your AWS account through Amazon Simple Notification Service. To send SMS messages with Amazon SNS in the AWS Region that you want, the Amazon Cognito user pool uses an AWS Identity and Access Management (IAM) role in your AWS account.

Type: SmsConfigurationType (p. 412) object

Required: No

SmsConfigurationFailure

The reason why the SMS configuration can't send the messages to your users.

This message might include comma-separated values to describe why your SMS configuration can't send messages to user pool end users.

InvalidSmsRoleAccessPolicyException

The AWS Identity and Access Management role that Amazon Cognito uses to send SMS messages isn't properly configured. For more information, see SmsConfigurationType.

SNSSandbox

The AWS account is in the SNS SMS Sandbox and messages will only reach verified end users. This parameter won’t get populated with SNSSandbox if the IAM user creating the user pool doesn’t have SNS permissions. To learn how to move your AWS account out of the sandbox, see Moving out of the SMS sandbox.

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

UserAttributeUpdateSettings

The settings for updates to user attributes. These settings include the property `AttributesRequireVerificationBeforeUpdate`, a user-pool setting that tells Amazon Cognito how to handle changes to the value of your users' email address and phone number attributes. For more information, see Verifying updates to to email addresses and phone numbers.

Type: UserAttributeUpdateSettingsType (p. 422) object

Required: No

UsernameAttributes

Specifies whether a user can use an email address or phone number as a username when they sign up.

Type: Array of strings

Valid Values: `phone_number | email`

Required: No

UsernameConfiguration

Case sensitivity of the username input for the selected sign-in option. For example, when case sensitivity is set to `False`, users can sign in using either "username" or "Username". This configuration is immutable once it has been set. For more information, see UsernameConfigurationType.

Type: UsernameConfigurationType (p. 427) object

Required: No

UserPoolAddOns

The user pool add-ons.

Type: UserPoolAddOnsType (p. 428) object

Required: No

UserPoolTags

The tags that are assigned to the user pool. A tag is a label that you can apply to user pools to categorize and manage them in different ways, such as by purpose, owner, environment, or other criteria.

Type: String to string map

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

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

Required: No

VerificationMessageTemplate

The template for verification messages.

Type: VerificationMessageTemplateType (p. 448) 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 V2
- AWS SDK for Ruby V3

UserType

A user profile in a Amazon Cognito user pool.

Contents

Attributes

A container with information about the user type attributes.

Type: Array of AttributeType (p. 359) 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. 393) 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 want 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. This can be one of the following:

- UNCONFIRMED - User has been created but not confirmed.
- CONFIRMED - User has been confirmed.

- EXTERNAL_PROVIDER - User signed in with a third-party IdP.
- ARCHIVED - User is no longer active.
- UNKNOWN - User status isn't known.
- RESET_REQUIRED - User is confirmed, but the user must request a code and reset their password before they can sign in.
- FORCE_CHANGE_PASSWORD - The user is confirmed and the user can sign in using a temporary password, but on first sign-in, the user must change their password to a new value before doing anything else.

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 V2
- AWS SDK for Ruby V3

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 template for email messages that Amazon Cognito sends to your users. You can set an `EmailMessage` template only if the value of EmailSendingAccount is `DEVELOPER`. When your EmailSendingAccount is `DEVELOPER`, your user pool sends email messages with your own Amazon SES configuration.

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. You can set an `EmailMessageByLink` template only if the value of EmailSendingAccount is `DEVELOPER`. When your EmailSendingAccount is `DEVELOPER`, your user pool sends email messages with your own Amazon SES configuration.

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. You can set an `EmailSubject` template only if the value of EmailSendingAccount is `DEVELOPER`. When your EmailSendingAccount is `DEVELOPER`, your user pool sends email messages with your own Amazon SES configuration.

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. You can set an `EmailSubjectByLink` template only if the value of EmailSendingAccount is `DEVELOPER`. When your EmailSendingAccount is `DEVELOPER`, your user pool sends email messages with your own Amazon SES configuration.

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 template for SMS messages that Amazon Cognito sends to your users.

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 V2
- AWS SDK for Ruby V3

Common Parameters

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

Action

The action to be performed.

Type: string

Required: Yes

Version

The API version that the request is written for, expressed in the format YYYY-MM-DD.

Type: string

Required: Yes

X-Amz-Algorithm

The hash algorithm that you used to create the request signature.

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

Type: string

Valid Values: `AWS4-HMAC-SHA256`

Required: Conditional

X-Amz-Credential

The credential scope value, which is a string that includes your access key, the date, the region you are targeting, the service you are requesting, and a termination string ("aws4_request"). The value is expressed in the following format: *access_key*/*YYYYMMDD*/*region*/*service*/aws4_request.

For more information, see Task 2: Create a String to Sign for Signature Version 4 in the *Amazon Web Services General Reference*.

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

Type: string

Required: Conditional

X-Amz-Date

The date that is used to create the signature. The format must be ISO 8601 basic format (YYYYMMDD'T'HHMMSS'Z'). For example, the following date time is a valid X-Amz-Date value: `20120325T120000Z`.

Condition: X-Amz-Date is optional for all requests; it can be used to override the date used for signing requests. If the Date header is specified in the ISO 8601 basic format, X-Amz-Date is not required. When X-Amz-Date is used, it always overrides the value of the Date header. For more information, see Handling Dates in Signature Version 4 in the *Amazon Web Services General Reference*.

Type: string

Required: Conditional

X-Amz-Security-Token

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

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

Type: string

Required: Conditional

X-Amz-Signature

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

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

Type: string

Required: Conditional

X-Amz-SignedHeaders

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

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

Type: string

Required: Conditional

Common Errors

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

AccessDeniedException

You do not have sufficient access to perform this action.

HTTP Status Code: 400

IncompleteSignature

The request signature does not conform to AWS standards.

HTTP Status Code: 400

InternalFailure

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

HTTP Status Code: 500

InvalidAction

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

HTTP Status Code: 400

InvalidClientTokenId

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

HTTP Status Code: 403

InvalidParameterCombination

Parameters that must not be used together were used together.

HTTP Status Code: 400

InvalidParameterValue

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

HTTP Status Code: 400

InvalidQueryParameter

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

HTTP Status Code: 400

MalformedQueryString

The query string contains a syntax error.

HTTP Status Code: 404

MissingAction

The request is missing an action or a required parameter.

HTTP Status Code: 400

MissingAuthenticationToken

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

HTTP Status Code: 403

MissingParameter

A required parameter for the specified action is not supplied.

HTTP Status Code: 400

NotAuthorized

You do not have permission to perform this action.

HTTP Status Code: 400

OptInRequired

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

HTTP Status Code: 403

RequestExpired

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

HTTP Status Code: 400

ServiceUnavailable

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

HTTP Status Code: 503

ThrottlingException

The request was denied due to request throttling.

HTTP Status Code: 400

ValidationError

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

HTTP Status Code: 400