AWS Identity and Access Management

CLI Reference

API Version 2010-05-08

Amazon Web Services

AWS Identity and Access Management: CLI Reference

Amazon Web Services

Welcome

This is the AWS Identity and Access Management (IAM) Command Line Reference. This guide provides descriptions of the IAM CLI as well as links to related content in the guide, Using Identity and Access Management.

IAM is a web service that enables Amazon Web Services (AWS) customers to manage users and user permissions under their AWS account. For more information about this product, go to AWS Identity and Access Management.

How Do I... ?

The following table lists links to information on how to get things done with AWS Identity and Access Management.

How Do I?	Relevant Resources
Get started with the command line tools	Getting the Command Line Tools (p. 2)
Get a list of common options used with all IAM commands	Common Command Options (p. 7)
Get a list of commands by function	Commands (p. 9)
Get developer tools	Developer Tools

Getting the Command Line Tools

To use the commands in this guide, you must get the command line interface (CLI). The interface is written in Java and includes shell scripts for both Windows and Linux/UNIX/Mac OSX.

You must have a Java SDK or JRE installed (version 1.6.x or later).

As a convention, all command line text in this guide is prefixed with a generic `PROMPT>` command line prompt. The actual command line prompt on your machine is likely to be different.

The CLI depends on three environment variables and a change to your system path. This section presents detailed steps for downloading the CLI and configuring your system to use it.

Note

Linux and Windows environment variables are reset whenever you close the command window. You might want to set your environment variables permanently. Consult the documentation for your version of Linux or Windows for more information on setting environment variables.

Note

The command line interface contains template files you can use to specify settings for AWS_CREDENTIAL_FILE and CLIENT_CONFIG_FILE. The templates are available at `$AWS_IAM_HOME/aws-credential.template` and `$AWS_IAM_HOME/client-config.template`, respectively.

Windows Users

Paths that contain a space must be wrapped in quotation marks, for example: `"C:\Program Files\Java"`.

Downloading the CLI

To download the CLI

1. Go to IAM Command Line Toolkit and click **Download**.
2. Save the file.

3. Extract the contents.

Installing and Configuring Java

The IAM CLI requires either a Java Development Kit (SDK) or a Java Runtime Environment (JRE). If you don't already have one, or have a version older than 1.6, download the latest version from the Java SE Downloads page.

After you download and install the Java SDK or JRE, you must create an environmental variable that points to where Java is installed.

To set the JAVA_HOME variable

1. Enter the path to the Java installation:

   - On Linux/UNIX, enter the following command:

     ```
     PROMPT> export JAVA_HOME=<path_to_your_Java_installation>
     ```

   - On Windows, enter the following command:

     ```
     PROMPT> set JAVA_HOME=<path_to_your_Java_installation>
     ```

2. Confirm that the variable is set:

   - On Linux/UNIX, enter the following command:

     ```
     PROMPT> ${JAVA_HOME}/bin/java -version
     ```

   - On Windows, enter the following command:

     ```
     PROMPT> %JAVA_HOME%\bin\java -version
     ```

   You will see output similar to the following:

   ```
   java version "1.6.0_21"
   Java(TM) SE Runtime Environment (build 1.6.0_21-b07)
   Java HotSpot(TM) Client VM (build 17.0-b17, mixed mode, sharing)
   ```

Setting Up the CLI

After you download and unzip the IAM CLI, you must create a variable for the location of the IAM CLI, and you must include IAM in your path.

To set the AWS_IAM_HOME environment variable

- On Linux/UNIX, enter the following command:

```
PROMPT> export AWS_IAM_HOME=<path_to_cli>
```

- On Windows, enter the following command:

```
PROMPT> set AWS_IAM_HOME=<path_to_cli>
```

To include IAM in your path

- On Linux/UNIX, enter the following command:

```
PROMPT> export PATH=$AWS_IAM_HOME/bin:$PATH
```

- On Windows, enter the following command:

```
PROMPT> set Path=%AWS_IAM_HOME%\bin;%Path%
```

Setting Up the Credentials File

You need to provide the CLI with the AWS Access Key ID and Secret Access Key for your AWS account. The CLI looks for these credentials in a file you create on your local system.

Tip

If you are the AWS account owner, you can get your AWS account's credentials by going to the AWS Security Credentials page. After you sign in, you can find the access keys located in the **Access Credentials** section of the page.

If you are a user under an AWS account, you can get your AWS security credentials from your account administrator.

To create the credential file

1. Use a text editor to create a text file that contains two lines: the first line lists the AWS Access Key ID, and the second line lists the Secret Access Key. For example:

```
AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE
AWSSecretKey=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
```

2. Save the file with any name you want (e.g., `account-key`).
3. Limit the file permissions to only the file owner (e.g., use `chmod 600` on the file if you're using Linux/UNIX).

Caution

Your Secret Access Key is a secret that only you and AWS (or your system administrator) should know. It is important to keep it confidential to protect your account. Store it securely in a safe place. Never include it in your requests to AWS, and never email it to anyone. Do not share it outside your organization, even if an inquiry appears to come from AWS or Amazon.com. No one who legitimately represents Amazon will ever ask you for your Secret Access Key.

After you set up the credentials file, you'll need to set the `AWS_CREDENTIAL_FILE` environment variable so that the CLI knows where to find your credentials.

To set the AWS_CREDENTIAL_FILE environment variable

1. Enter the path to the credentials file:

   - On Linux/UNIX, enter following command:

   ```
   PROMPT> export AWS_CREDENTIAL_FILE=<path_and_filename_of_credential_file>
   ```

   - On Windows, enter the following command:

   ```
   PROMPT> set AWS_CREDENTIAL_FILE=<path_and_filename_of_credential_file>
   ```

2. Run one of the commands in help mode to verify that your setup works properly:

   ```
   PROMPT> iam-usercreate -h
   ```

   You should see the help for the `iam-usercreate` command, which looks similar to this:

   ```
   Creates a new user in your account. You can also optionally add the user
   to one or more groups, and create an access key for the user.
   iam-usercreate [options...] arguments...

    --aws-credential-file CREDENTIALFILE : path to the file containing your
   AWS credentials
    -d (--debug)                         : enable debug logging
    -g GROUPS                            : add user to group(s)
    -h                                   : print out this message
    -k                                   : create a key for the user
    -p PATH                              : the path of the user, defaults to
   /
    -u USERNAME                          : the name of the user
    -v VERBOSE                           : print out the newly created user's
    arn and guid
   ```

Setting Up the Proxy Configuration File

If your connection uses a proxy server (this is not common), you need to provide the CLI with the proxy settings. If you do not use a proxy server, then you can skip this step.

If you use a proxy server, the CLI looks for these configuration settings in a file you create on your local system.

To create the configuration file

1. Use a text editor to create a text file that contains the following lines:

   ```
   ClientProxyHost=<your proxy server name>
   ClientProxyPort=<your proxy server port number>
   ClientProxyUsername=<your proxy user name>
   ClientProxyPassword=<your proxy password>
   ```

2. Save the file with any name you want (e.g., `myconfig.txt`).
3. Limit the file permissions to only the file owner (e.g., use `chmod 600` on the file if you're using Linux/UNIX).

After you set up the configuration file, you'll need to set the `CLIENT_CONFIG_FILE` environment variable so that the CLI knows where to find your proxy settings.

To set the CLIENT_CONFIG_FILE environment variable

- On Linux/UNIX, enter following command:

  ```
  PROMPT> export CLIENT_CONFIG_FILE=<path_and_filename_of_configuration_file>
  ```

- On Windows, enter the following command:

  ```
  PROMPT> set CLIENT_CONFIG_FILE=<path_and_filename_of_configuration_file>
  ```

You're now ready to use the IAM command line interface.

Common Command Options

This section describes options common to all IAM commands.

Options

All of the commands in the CLI accept the optional parameters described in the following table.

Option	Description
`--aws-credential-file` *value*	Path to the file containing your AWS credentials. This value can be stored in the AWS_CREDENTIAL_FILE environment variable. Example: `--aws-credential-file c:\AWS\mycredentials.txt`
`--client-config-file` *value*	Path to the file containing your proxy server settings. If you are behind a proxy server and you cannot make calls to IAM directly, you can use this option to specify a configuration file that contains your proxy server settings. This value can be stored in the CLIENT_CONFIG_FILE environment variable. Example: `--client-config-file c:\AWS\myconfig.txt`
`-d or --debug`	Enables debug logging.
`-h`	Prints help information for the command.
`--url` *value*	Override the URL for the service call with the value supplied. This value is set using the AWS_IAM_URL environment variable.

Note

When you use the `--aws-credential-file` or `--client-config-file` option on the command line, you override the option value stored as an environment variable.

Note

The command line interface contains template files you can use to specify settings for `--aws-credential-file` and `--client-config-file`. The templates are available at `$AWS_IAM_HOME/aws-credential.template` and `$AWS_IAM_HOME/client-config.template`, respectively.

Commands

This section describes commands you can perform on IAM entities. For more information about these entities, refer to Using AWS Identity and Access Management.

Commands for Groups

- iam-groupaddpolicy (p. 12)
- iam-groupadduser (p. 15)
- iam-groupcreate (p. 16)
- iam-groupdel (p. 18)
- iam-groupdelpolicy (p. 20)
- iam-grouplistbypath (p. 21)
- iam-grouplistpolicies (p. 23)
- iam-grouplistusers (p. 25)
- iam-groupmod (p. 27)
- iam-groupremoveuser (p. 29)
- iam-groupuploadpolicy (p. 30)

Commands for Users

- iam-useraddcert (p. 62)
- iam-useraddkey (p. 64)
- iam-useraddloginprofile (p. 66)
- iam-useraddpolicy (p. 67)
- iam-userchangepassword (p. 70)
- iam-usercreate (p. 71)
- iam-userdeactivatemfadevice (p. 73)
- iam-userdel (p. 74)
- iam-userdelcert (p. 76)
- iam-userdelkey (p. 77)
- iam-userdelloginprofile (p. 78)
- iam-userdelpolicy (p. 79)
- iam-userenablemfadevice (p. 80)
- iam-usergetattributes (p. 82)

- iam-usergetloginprofile (p. 84)
- iam-userlistbypath (p. 85)
- iam-userlistcerts (p. 87)
- iam-userlistgroups (p. 89)
- iam-userlistkeys (p. 91)
- iam-userlistmfadevices (p. 93)
- iam-userlistpolicies (p. 95)
- iam-usermod (p. 97)
- iam-usermodcert (p. 99)
- iam-usermodkey (p. 101)
- iam-usermodloginprofile (p. 103)
- iam-userresyncmfadevice (p. 104)
- iam-useruploadpolicy (p. 106)

Commands for Roles

- iam-roleaddpolicy (p. 46)
- iam-rolecreate (p. 48)
- iam-roledel (p. 50)
- iam-roledelpolicy (p. 52)
- iam-rolegetattributes (p. 53)
- iam-rolelistbypath (p. 54)
- iam-rolelistpolicies (p. 56)
- iam-roleupdateassumepolicy (p. 58)
- iam-roleuploadpolicy (p. 60)

Commands for Instance Profiles

- iam-instanceprofileaddrole (p. 32)
- iam-instanceprofilecreate (p. 34)
- iam-instanceprofiledel (p. 36)
- iam-instanceprofilegetattributes (p. 38)
- iam-instanceprofilelistbypath (p. 40)
- iam-instanceprofilelistforrole (p. 42)
- iam-instanceprofileremoverole (p. 44)

Commands for Server Certificates

- iam-servercertdel (p. 108)
- iam-servercertgetattributes (p. 109)
- iam-servercertlistbypath (p. 110)
- iam-servercertmod (p. 112)
- iam-servercertupload (p. 114)

Commands for AWS Accounts

- iam-accountaliascreate (p. 116)

- iam-accountaliasdelete (p. 117)
- iam-accountaliaslist (p. 118)
- iam-accountdelpasswordpolicy (p. 120)
- iam-accountgetpasswordpolicy (p. 121)
- iam-accountgetsummary (p. 122)
- iam-accountmodpasswordpolicy (p. 123)

Commands for MFA

- iam-userdeactivatemfadevice (p. 73)
- iam-userenablemfadevice (p. 80)
- iam-userlistmfadevices (p. 93)
- iam-userresyncmfadevice (p. 104)
- iam-virtualmfadevicecreate (p. 125)
- iam-virtualmfadevicedel (p. 127)
- iam-virtualmfadevicelist (p. 128)

iam-groupaddpolicy

Description

Creates a policy based on the information you provide and attaches the policy to the specified group. Use this command if you need a simple policy with no conditions, and you don't want to write the policy yourself. If you need a policy with conditions, you must write the policy yourself and upload it with iam-groupuploadpolicy (p. 30). For information about the contents of policies, refer to *Using AWS Identity and Access Management*.

You can add only a limited number of policies to a group. For more information, see Appendix A: Limitations on IAM Entities (p. 131).

⚠ **Important**

This command overwrites any existing policy with the same name and same entity associations.

Syntax

```
iam-groupaddpolicy -g GROUPNAME -p POLICYNAME -e EFFECT {-a ACTION ...} {-r
AMAZON RESOURCE NAME ...} [-o]
```

Options

Name	Description	Required
`-g` *GROUPNAME*	Name of the group the policy is for. Type: String Default: None	Yes
`-p` *POLICYNAME*	Name you want to assign the policy. Type: String Default: None	Yes
`-e` *EFFECT*	The value for the policy's `Effect` element. Specifies whether the policy results in an allow or a deny. For more information about policies and their contents, refer to *Using AWS Identity and Access Management*. Type: String Valid Values: `Allow` | `Deny` Default: None	Yes

Name	Description	Required
`-a` *`ACTION`*	The value for the policy's `Action` element. Specifies the service and action you want to allow or deny permission to. For example: `-a iam:ListAccessKeys`. You can use wildcards, and you can specify more than one `-a Action` option in the request. The following example specifies all the IAM actions related to access keys or signing certificates: `-a iam:*AccessKey* -a iam:*SigningCertificate*` Type: String Default: None	Yes
`-r` *`AMAZON RESOURCE NAME`*	The value for the policy's `Resource` element. Specifies the Amazon Resource Name (ARN) for the resource (or resources) the policy applies to. You can use wildcards, and you can specify more than one `-r RESOURCE` option in the request. Quotation marks are required if you're just specifying * as the resource. The following example specifies all the resources in the AWS account: `-r "*"` . The following example specifies all groups in the AWS account: `-r arn:aws:iam::123456789012:group/*` Type: String Default: None	Yes
`-o`	Causes the output to include the JSON policy document that IAM created for you.	No

Output

If the command is successful, the output is empty. Exception: if you specified the `-o` option, the output includes the JSON policy document.

Example

The following example request adds (or updates) the policy named AdminRoot for the group named Admins. The `-o` option causes the output to include the JSON policy document we construct for you based on the options you provided.

```
PROMPT> iam-groupaddpolicy -g Admins -p AdminRoot -e Allow -a "*" -r "*" -o

{"Version":"2008-10-17","Statement":[{"Effect":"Allow","Action":["*"],"Re
source":["*"]}]}
```

Related Commands

- iam-groupdelpolicy (p. 20)
- iam-grouplistpolicies (p. 23)

- iam-groupuploadpolicy (p. 30)

iam-groupadduser

Description

Adds one or more users to a group.

Syntax

```
iam-groupadduser -g GROUPNAME [-u USERNAME ...]
```

Options

Name	Description	Required
`-g` *GROUPNAME*	Name of the group to add the user to. Type: String Default: None	Yes
`-u` *USERNAME*	Name of the user to add. To add multiple users, you can repeat this option. Type: String Default: None	Yes

Output

If the command is successful, the output is empty.

Example

The following example adds the users Jill and Jason to the group called Test.

```
PROMPT> iam-groupadduser -g Test -u Jill -u Jason
```

Related Commands

- iam-groupremoveuser (p. 29)
- iam-grouplistusers (p. 25)
- iam-grouplistbypath (p. 21)

iam-groupcreate

Description

Creates a new empty group. An AWS account can have only a limited number of groups. For more information, see Appendix A: Limitations on IAM Entities (p. 131).

Syntax

```
iam-groupcreate -g GROUPNAME [-p PATH] [-v]
```

Options

Name	Description	Required
`-g GROUPNAME`	Name of the group to create. Do not include the path in this value. Type: String Constraints: See Appendix A: Limitations on IAM Entities (p. 131) Default: None	Yes
`-p PATH`	Path to the group. For more information about paths, go to Identifiers for IAM Entities in *Using AWS Identity and Access Management*. If you don't want the group to have a path, set to `/`. Type: String Constraints: See Appendix A: Limitations on IAM Entities (p. 131) Default: /	No
`-v`	Causes the response to include the newly created group's ARN and GUID. For more information about ARNs and GUIDs, go to Identifiers for IAM Entities in *Using AWS Identity and Access Management*. Type: String Default: None	No

Output

If the command is successful, the output is empty. Exception: If you use the `-v` option, the response includes the group's ARN and GUID.

Examples

The following example creates a new group called Admins with no path. You could omit the -p option and get the same result.

```
PROMPT> iam-groupcreate -g Admins -p / -v
arn:aws:iam::123456789012:group/Admins
AIDACKCEVSQ6C2EXAMPLE
```

The following example creates a new group called Managers with a path of /division_abc/subdivision_xyz/product_1234/.

```
PROMPT> iam-groupcreate -g Managers -p /division_abc/subdivision_xyz/
product_1234/ -v
arn:aws:iam::123456789012:group/division_abc/subdivision_xyz/product_1234/Man
agers
AIDGPMS9RO4H3FEXAMPLE
```

Related Commands

- iam-groupdel (p. 18)
- iam-grouplistbypath (p. 21)
- iam-groupmod (p. 27)

iam-groupdel

Description

Deletes a group from your AWS account. If using this command only with the `-g` option, the group must be empty and have no attached policies.

If you want to delete the group, delete the users from the group, and delete its attached policies all at once, you can use the `-r` option to recursively delete the group. Recursively deleting the group automatically removes user associations from the group and deletes any attached policies along with the group.

⚠ **Important**

Use the `-r` option with caution. Before performing a recursive delete, to ensure you are not deleting anything you don't want to, use the `-p` option along with the `-r` option to list all the users in the group and any attached policies without actually performing the recursive deletion.

Syntax

```
iam-groupdel -g GROUPNAME [ -r [-p] ]
```

Options

Name	Description	Required
`-g` *GROUPNAME*	Name of the group to delete. Type: String Default: None	Yes
`-r`	Removes any users from the group and deletes any attached policies while deleting the group. Type: String Default: None	Optional
`-r -p`	Returns a list of associated users and policies, without actually deleting the group. Use this before using `-r` to ensure you are not deleting anything you don't want deleted. The `-p` option indicates *pretend mode*; use only with the `-r` option. Type: String Constraints: You can apply -p only together with -r. Default: None	Optional

Output

If the deletion is successful, the output is empty.

Examples

The following example deletes the group called Test. This example assumes the group is empty and has no policies attached.

```
PROMPT> iam-groupdel -g Test
```

The following example shows the user associations and the policies that would be deleted if you were to recursively delete the Test group.

```
PROMPT> iam-groupdel -g Test -r -p
users
     arn:aws:iam::123456789012:user/division_abc/subdivision_xyz/Susan
     arn:aws:iam::123456789012:user/division_abc/subdivision_xyz/John
policies
     TestGroupPolicy
```

The following example recursively deletes the Test group.

```
PROMPT> iam-groupdel -g Test -r
```

Related Commands

- iam-groupcreate (p. 16)
- iam-grouplistbypath (p. 21)
- iam-grouplistusers (p. 25)
- iam-groupmod (p. 27)

iam-groupdelpolicy

Description

Removes a policy from the specified group.

Syntax

iam-groupdelpolicy -g *GROUPNAME* -p *POLICYNAME*

Options

Name	Description	Required
-g *GROUPNAME*	Name of the group the policy is attached to. Type: String Default: None	Yes
-p *POLICYNAME*	Name of the policy document to delete. Type: String Default: None	Yes

Output

If the command is successful, the output is empty.

Example

The following example request deletes the policy named TestPolicy from the group named Managers.

```
PROMPT> iam-groupdelpolicy -g Managers -p TestPolicy
```

Related Commands

- iam-groupaddpolicy (p. 12)
- iam-grouplistpolicies (p. 23)
- iam-groupuploadpolicy (p. 30)

iam-grouplistbypath

Description

Lists all the groups in the AWS account, or lists the groups in the AWS account that have the specified path prefix. If no groups exist to list, the action still succeeds. You can paginate the results using the `MaxItems` and `Marker` options.

Syntax

```
iam-grouplistbypath [-p PATH]
```

Options

Name	Description	Required
`-p` *`PATH`*	The path prefix for filtering the results. For example, `/division_abc/subdivision_xyz/` would get all groups whose path starts with /division_abc/subdivision_xyz/. Type: String Default: / Condition: Provide this option only if you want to list the groups with a specific path prefix.	Conditional
`-i` *`MAXITEMS`*	Use this option only when paginating results to indicate the maximum number of items you want in the response. If there are additional items beyond the maximum you specify, the `IsTruncated` response element is `true`. Type: String Default: None	No
`-m` *`MARKER`*	Use this only when paginating results, and only in a subsequent request after you've received a response where the results are truncated. Set it to the value of the Marker element in the response you just received. Type: String Default: None	No

Output

The output lists the Amazon Resource Names (ARNs) that have the specified path prefix.

Examples

The following example lists all the groups in the AWS account.

```
PROMPT> iam-grouplistbypath
groups
   arn:aws:iam::123456789012:group/Admins
   arn:aws:iam::123456789012:group/division_abc/subdivision_xyz/product_1234/en
gineering/Test
   arn:aws:iam::123456789012:group/division_abc/subdivision_xyz/product_1234/Man
agers
```

The following example lists all the groups whose path starts with /division_abc/subdivision_xyz/.

```
PROMPT> iam-grouplistbypath -p /division_abc/subdivision_xyz/
groups
   arn:aws:iam::123456789012:group/division_abc/subdivision_xyz/product_1234/en
gineering/Test
   arn:aws:iam::123456789012:group/division_abc/subdivision_xyz/product_1234/Man
agers
```

Related Commands

- iam-groupcreate (p. 16)
- iam-groupdel (p. 18)
- iam-grouplistusers (p. 25)
- iam-groupmod (p. 27)

iam-grouplistpolicies

Description

Lists one specific policy or all the policies attached to the specified group. If no policies are attached to the group, the action still succeeds. You can paginate the results using the `MaxItems` and `Marker` options.

Syntax

```
iam-grouplistpolicies -g GROUPNAME [-p POLICYNAME] [-v]
```

Options

Name	Description	Required
`-g` *GROUPNAME*	Name of the group the policy is attached to. Type: String Default: None	Yes
`-p` *POLICYNAME*	Name of the policy document to display. Type: String Default: None	No
`-v`	Displays the contents of the resulting policies (in addition to the policy names). Type: String	No
`-i` *MAXITEMS*	Use this option only when paginating results to indicate the maximum number of items you want in the response. If there are additional items beyond the maximum you specify, the `IsTruncated` response element is `true`. Type: String Default: None	No
`-m` *MARKER*	Use this only when paginating results, and only in a subsequent request after you've received a response where the results are truncated. Set it to the value of the Marker element in the response you just received. Type: String Default: None	No

Output

The output contains the contents of the specific policy you requested, or it contains the names of the policies attached to the specified group (and optionally the contents of each).

Examples

The following example request displays the policy named AdminRoot, which is attached to the group named Admins.

```
PROMPT> iam-grouplistpolicies -g Admins -p AdminRoot -v
{"Version":"2008-10-17","Statement":[{"Effect":"Allow","Action":["*"],"Re
source":["*"]}]}
```

The following example request displays the names of all the policies attached to the group named Managers. You could optionally have the output display the contents of the policies by including the `-v` option.

```
PROMPT> iam-grouplistpolicies -g Managers
KeyPolicy
AnotherManagerGroupPolicy
```

Related Commands

- iam-groupaddpolicy (p. 12)
- iam-groupdelpolicy (p. 20)
- iam-groupuploadpolicy (p. 30)

iam-grouplistusers

Description

Lists all the users in a group. You can paginate the results using the `MaxItems` and `Marker` options.

Syntax

`iam-grouplistusers [-g` ***`GROUPNAME`*****`]`**

Options

Name	Description	Required
`-g` *`GROUPNAME`*	The group for filtering the results. Type: String Default: None	Yes
`-i` *`MAXITEMS`*	Use this option only when paginating results to indicate the maximum number of items you want in the response. If there are additional items beyond the maximum you specify, the `IsTruncated` response element is `true`. Type: String Default: None	No
`-m` *`MARKER`*	Use this only when paginating results, and only in a subsequent request after you've received a response where the results are truncated. Set it to the value of the Marker element in the response you just received. Type: String Default: None	No

Output

The output lists the Amazon Resource Names (ARNs) for the specified group.

Example

The following example lists the users in the Admins group.

```
PROMPT> iam-grouplistusers -g Admins
   arn:aws:iam::123456789012:group/Admins
      users
      arn:aws:iam::123456789012:user/division_abc/subdivision_xyz/Bob
      arn:aws:iam::123456789012:user/division_abc/subdivision_xyz/John
```

Related Commands

- iam-grouplistbypath (p. 21)
- iam-groupcreate (p. 16)
- iam-groupdel (p. 18)
- iam-groupmod (p. 27)

iam-groupmod

Description

Changes the group's name or path (or both).

Important

You need to understand the implications of changing a group's path or name. For more information, see Renaming Users and Groups in *Using AWS Identity and Access Management*.

Note

To change a group name the requester must have appropriate permissions on both the source object and the target object. For example, to change Managers to MGRs, the entity making the request must have permission on Managers and MGRs, or must have permission on all (*). For more information about permissions, see Permissions and Policies.

Syntax

```
iam-groupmod -g GROUPNAME [-n NEWGROUPNAME] [-p NEWPATH]
```

Options

Name	Description	Required
`-g` *`GROUPNAME`*	Name of the group to update. If you're changing the group's name, this is the original name. Type: String Default: None	Yes
`-n` *`NEWGROUPNAME`*	New name for the group. Type: String Default: None Condition: Only include this if changing the group's name.	Conditional
`-p` *`NEWPATH`*	New path for the group. Type: String Default: None Condition: Only include this if changing the group's path.	Conditional

Output

If the request is successful, the output is empty.

Examples

The following example changes the name of the group from Test to Test_1.

```
PROMPT> iam-groupmod -g Test -n Test_1
```

The following example changes the path for Test to /division_abc/subdivision_xyz/test/.

```
PROMPT> iam-groupmod -g Test -p /division_abc/subdivision_xyz/test/
```

Related Commands

- iam-groupcreate (p. 16)
- iam-groupdel (p. 18)
- iam-grouplistbypath (p. 21)
- iam-grouplistusers (p. 25)

iam-groupremoveuser

Description

Removes one or more users from a group.

Syntax

```
iam-groupremoveuser -g GROUPNAME [-u USERNAME ...]
```

Options

Name	Description	Required
`-g` *GROUPNAME*	Name of the group to remove the user from. Type: String Default: None	Yes
`-u` *USERNAME*	Name of the user to remove from the group. To remove multiple users, you can repeat this option. Type: String Default: None	Yes

Output

If the command is successful, the output is empty.

Example

The following example removes the users Jill and Jason from the group called Test.

```
PROMPT> iam-groupremoveuser -g Test -u Jill -u Jason
```

Related Commands

- iam-groupadduser (p. 15)
- iam-grouplistusers (p. 25)
- iam-grouplistbypath (p. 21)

iam-groupuploadpolicy

Description

Takes a policy you've written and attaches it to the specified group. If a policy with that name is already attached to the group, it's overwritten with the new one. The command accepts either a string representing the policy, or a file containing the policy. For information about the contents of policies, refer to *Using AWS Identity and Access Management*.

You can add only a limited number of policies to a group. For more information, see Appendix A: Limitations on IAM Entities (p. 131).

Syntax

```
iam-groupuploadpolicy -g GROUPNAME -p POLICYNAME [-f POLICYDOCUMENTFILE | -o POLICYDOCUMENT]
```

Options

Name	Description	Required
`-g` *`GROUPNAME`*	Name of the group the policy is for. Type: String Default: None	Yes
`-p` *`POLICYNAME`*	Name you want to assign the policy. Type: String Default: None	Yes
`-f` *`POLICYDOCUMENTFILE`*	Path and name of the file containing the policy. Type: String Condition: Either `-f POLICYDOCUMENTFILE` or `-o POLICYDOCUMENT` is required. If you use both options together, IAM returns an error. Default: None	Conditional
`-o` *`POLICYDOCUMENT`*	The policy (a JSON text string). Type: String Condition: Either `-f POLICYDOCUMENTFILE` or `-o POLICYDOCUMENT` is required. If you use both options together, IAM returns an error. Default: None	Conditional

Output

If the command is successful, the output is empty.

Example

The following example adds (or updates) the policy named AdminRoot for the group named Admins. The policy is uploaded as a text string.

```
PROMPT> iam-groupuploadpolicy -g Admins -p AdminRoot -o {"Statement":[{"Ef
fect":"Allow","Action":"*","Resource":"*"}]}
```

The following example adds (or updates) the policy named AdminRoot for the group named Admins. The policy is uploaded as a text file.

```
PROMPT> iam-groupuploadpolicy -g Admins -p AdminRoot -f C:\Policies\Admin
Root_file.txt
```

Related Commands

- iam-groupaddpolicy (p. 12)
- iam-groupdelpolicy (p. 20)
- iam-grouplistpolicies (p. 23)

iam-instanceprofileaddrole

Description

Adds a role to an instance profile.

Note

Currently, you can add only one role to an instance profile.

For more information about instance profiles, see About Instance Profiles in *Using AWS Identity and Access Management*. Working with roles is described in Working with Roles.

Syntax

```
iam-instanceprofileaddrole -r ROLENAME -s INSTANCEPROFILENAME
```

Options

Name	Description	Required
-r *ROLENAME*	Name of the role to add to the instance profile. Type: String Default: None	Yes
-s *INSTANCEPROFILENAME*	The instance profile to which you are adding the role. Type: String Default: None	Yes

Output

If the command is successful, the output is empty.

Example

The following example adds the role named *myrole* to the instance profile named *myinstanceprofile*.

```
PROMPT> iam-instanceprofileaddrole -s myinstanceprofile -r myrole
```

Related Commands

- iam-instanceprofilecreate (p. 34)
- iam-instanceprofiledel (p. 36)
- iam-instanceprofilegetattributes (p. 38)
- iam-instanceprofilelistbypath (p. 40)
- iam-instanceprofilelistforrole (p. 42)

- iam-instanceprofileremoverole (p. 44)

iam-instanceprofilecreate

Description

Creates a new instance profile in your AWS account. Optionally adds a role to the instance profile.

For more information about instance profiles, see About Instance Profiles in *Using AWS Identity and Access Management*. Working with roles is described in Working with Roles.

Syntax

```
iam-instanceprofilecreate [-p PATH | -r ROLENAME] -s INSTANCEPROFILENAME
```

Options

Name	Description	Required
`-p` *`PATH`*	Path to the instance profile. For more information about paths, go to Identifiers for IAM Entities in *Using AWS Identity and Access Management*. If you don't want the instance profile to have a path, set to /. Type: String Constraints: See Appendix A: Limitations on IAM Entities (p. 131). Default: /	No
`-r` *`ROLENAME`*	The name of the role to add to the instance profile. Type: String Default: None	Optional
`-s` *`INSTANCEPROFILENAME`*	Name of the instance profile. Type: String Constraints: See Appendix A: Limitations on IAM Entities (p. 131). Default: None	Yes

Output

The output lists the Amazon Resource Name (ARN) for the instance profile. For more information about ARNs, see ARNs in *Using AWS Identity and Access Management*.

Example

The following example creates an instance profile named *myinstanceprofile* with no path. You could omit the `-p` option and get the same result.

```
PROMPT> iam-instanceprofilecreate -s myinstanceprofile -p /
arn:aws:iam::123456789012:instance-profile/myinstanceprofile
```

The following example creates an instance profile named *myinstanceprofile* with a path of /division_abc/subdivision_xyz/, and adds a previously created role named *myrole*.

```
PROMPT> iam-instanceprofilecreate -s myinstanceprofile -p /division_abc/subdi
vision_xyz/ -r myrole
arn:aws:iam::123456789012:instance-profile/division_abc/subdivision_xyz/myin
stanceprofile
```

Related Commands

- iam-instanceprofileaddrole (p. 32)
- iam-instanceprofiledel (p. 36)
- iam-instanceprofilegetattributes (p. 38)
- iam-instanceprofilelistbypath (p. 40)
- iam-instanceprofilelistforrole (p. 42)
- iam-instanceprofileremoverole (p. 44)

iam-instanceprofiledel

Description

Deletes an instance profile.

Caution

Make sure you do not have any Amazon EC2 instances running with the role or instance profile you are about to delete. Deleting a role or instance profile that is associated with a running instance will break any applications running on the instance.

For more information about instance profiles, go to About Instance Profiles in *Using AWS Identity and Access Management*.

Syntax

```
iam-instanceprofiledel -s INSTANCEPROFILENAME [ -r [-p] ]
```

Options

Name	Description	Required
`-s` *`INSTANCEPROFILENAME`*	The name of the instance profile to delete. Type: String	Yes
`-r`	Deletes the associated roles along with the instance profile. Type: String Default: None	Optional
`-r -p`	Returns the roles that would be deleted, without actually recursively deleting the instance profile or the roles. Use this option before using `-r` to ensure you are not deleting any roles you don't want to. The `-p` option indicates *pretend mode*; use only with the `-r` option. Type: String Constraints: You can apply -p only together with -r. Default: None	Optional

Output

If successful, the output is empty.

Example

The following example deletes the instance profile named *myinstanceprofile*. This example assumes the instance profile doesn't have roles attached.

```
PROMPT> iam-instanceprofiledel -s myinstanceprofile
```

The following example shows the roles associated with the instance profile that would be deleted if you were to recursively delete *myinstanceprofile*.

```
PROMPT> iam-instanceprofiledel -s myinstanceprofile -r -p
roles
    arn:aws:iam::123456789012:role/myrole
```

The following example recursively deletes *myinstanceprofile*.

```
PROMPT> iam-instanceprofiledel -s myinstanceprofile -r
```

Related Commands

- iam-instanceprofileaddrole (p. 32)
- iam-instanceprofilecreate (p. 34)
- iam-instanceprofilegetattributes (p. 38)
- iam-instanceprofilelistbypath (p. 40)
- iam-instanceprofilelistforrole (p. 42)
- iam-instanceprofileremoverole (p. 44)

iam-instanceprofilegetattributes

Description

Returns information about an instance profile. For more information about instance profiles, go to About Instance Profiles in *Using AWS Identity and Access Management*.

Syntax

iam-instanceprofilegetattributes -s *INSTANCEPROFILENAME* [-r]

Options

Name	Description	Required
-s *INSTANCEPROFILENAME*	The name of the instance profile you want to get information for. Type: String	Yes
-r	Lists associated roles along with the instance profile information. Type: String Default: None	Optional

Output

The output lists the instance profile Amazon Resource Name (ARN) and the GUID. For information about ARNs, go to ARNs in *Using AWS Identity and Access Management*.

Example

The following example lists the ARN and GUID for the instance profile named *myinstanceprofile*.

```
PROMPT> iam-instanceprofilegetattributes -s myinstanceprofile
```

The following example lists the ARN and GUID for the instance profile named *myinstanceprofile*, as well as the roles associated with the instance profile.

```
PROMPT> iam-instanceprofilegetattributes -s myinstanceprofile -r
arn:aws:iam::123456789012:instance-profile/division_abc/subdivision_xyz/myin
stanceprofile
AIPAJN3DDLPRMREXAMPLE
arn:aws:iam::123456789012:role/myrole
```

Related Commands

- iam-instanceprofileaddrole (p. 32)

- iam-instanceprofilecreate (p. 34)
- iam-instanceprofiledel (p. 36)
- iam-instanceprofilelistbypath (p. 40)
- iam-instanceprofilelistforrole (p. 42)
- iam-instanceprofileremoverole (p. 44)

iam-instanceprofilelistbypath

Description

Lists all the instance profiles in the AWS account, or lists the instance profiles in the AWS account that have the specified path prefix. If no instance profiles exist to list, the action still succeeds. You can paginate the results using the `MaxItems` and `Marker` options.

For more information about instance profiles, go to About Instance Profiles in *Using AWS Identity and Access Management*.

Syntax

```
iam-instanceprofilelistbypath [-p PATHPREFIX]
```

Options

Name	Description	Required
`-p` *`PATHPREFIX`*	The path prefix for filtering the results. For example, `/division_abc/subdivision_xyz/` would get all instance profiles whose path starts with /division_abc/subdivision_xyz/. Type: String Default: / Condition: Use this option only if you want to list the instance profiles that have a specific path prefix.	Conditional
`-i` *`MAXITEMS`*	Use this option only when paginating results to indicate the maximum number of items you want in the response. If there are additional items beyond the maximum you specify, the `IsTruncated` response element is `true`. Type: String Default: None	No
`-m` *`MARKER`*	Use this only when paginating results, and only in a subsequent request after you've received a response where the results are truncated. Set it to the value of the Marker element in the response you just received. Type: String Default: None	No

Output

The output lists the Amazon Resource Names (ARNs) for the instance profiles that have the specified path prefix. For information about ARNs, go to ARNs in *Using AWS Identity and Access Management*.

Example

The following example lists all the instance profiles in the AWS account.

```
PROMPT> iam-instanceprofilelistbypath
arn:aws:iam::123456789012:instance-profile/myinstanceprofile1
arn:aws:iam::123456789012:instance-profile/myinstanceprofile2
arn:aws:iam::123456789012:instance-profile/division_abc/subdivision_xyz/myin
stanceprofile3
IsTruncated: false
```

The following example lists all the instance profiles whose path starts with /division_abc/subdivision_xyz/.

```
PROMPT> iam-instanceprofilelistbypath -p /division_abc/subdivision_xyz/
arn:aws:iam::123456789012:instance-profile/division_abc/subdivision_xyz/myin
stanceprofile3
IsTruncated: false
```

Related Commands

- iam-instanceprofileaddrole (p. 32)
- iam-instanceprofilecreate (p. 34)
- iam-instanceprofiledel (p. 36)
- iam-instanceprofilegetattributes (p. 38)
- iam-instanceprofilelistforrole (p. 42)
- iam-instanceprofileremoverole (p. 44)

iam-instanceprofilelistforrole

Description

Lists all the instance profiles associated with a given role. You can paginate the results using the `MaxItems` and `Marker` options.

For more information about instance profiles, see About Instance Profiles in *Using AWS Identity and Access Management*. Working with roles is described in Working with Roles.

Syntax

`iam-instanceprofilelistforroles -r` ***`ROLENAME`***

Options

Name	Description	Required
`-r` *`ROLENAME`*	The role that you want to list the associated instance profiles for. Type: String	Yes
`-i` *`MAXITEMS`*	Use this option only when paginating results to indicate the maximum number of items you want in the response. If there are additional items beyond the maximum you specify, the `IsTruncated` response element is `true`. Type: String Default: None	No
`-m` *`MARKER`*	Use this only when paginating results, and only in a subsequent request after you've received a response where the results are truncated. Set it to the value of the Marker element in the response you just received. Type: String Default: None	No

Output

The output lists the Amazon Resource Names (ARNs) for the instance profiles that are associated with the specified role. For more information about ARNs, go to ARNs in *Using AWS Identity and Access Management*.

Example

The following example lists all the instance profiles associated with the role *myrole*.

```
PROMPT> iam-instanceprofilelistforrole -r myrole
arn:aws:iam::123456789012:instance-profile/division_abc/subdivision_xyz/myin
stanceprofile
IsTruncated: false
```

Related Commands

- iam-instanceprofileaddrole (p. 32)
- iam-instanceprofilecreate (p. 34)
- iam-instanceprofiledel (p. 36)
- iam-instanceprofilegetattributes (p. 38)
- iam-instanceprofilelistbypath (p. 40)
- iam-instanceprofileremoverole (p. 44)

iam-instanceprofileremoverole

Description

Removes a specified role from a specified instance profile.

Note

Removing a role from an instance profile breaks applications using the role on your Amazon EC2 instance.

For more information about instance profiles, see About Instance Profiles in *Using AWS Identity and Access Management*. Working with roles is described in Working with Roles.

Syntax

```
iam-instanceprofileremoverole -r ROLENAME -s INSTANCEPROFILENAME
```

Options

Name	Description	Required
-r *ROLENAME*	The name of the role that you want to remove from the specified instance profile. Type: String	Yes
-s *INSTANCEPROFILENAME*	The name of the instance profile from which you want to remove the specified role. Type: String	Yes

Output

If the command is successful, the output is empty.

Example

The following example removes the role named *myrole* from the instance profile named *myinstanceprofile*.

```
PROMPT> iam-instanceprofileremoverole -r myrole -s myinstanceprofile
```

Related Commands

- iam-instanceprofileaddrole (p. 32)
- iam-instanceprofilecreate (p. 34)
- iam-instanceprofiledel (p. 36)
- iam-instanceprofilegetattributes (p. 38)
- iam-instanceprofilelistbypath (p. 40)

- iam-instanceprofilelistforrole (p. 42)

iam-roleaddpolicy

Description

Creates a policy based on the information you provide and attaches the policy to the specified role. Use this command if you need a simple policy with no conditions, and you don't want to write the policy yourself. If you need a policy with conditions, you must write the policy yourself and upload it with iam-roleuploadpolicy (p. 60). For information about policy size limits, see Appendix A: Limitations on IAM Entities (p. 131).

For information about how to write policies and how policies work, go to Permissions and Policies in *Using AWS Identity and Access Management*.

Syntax

```
iam-roleaddpolicy -r ROLENAME -p POLICYNAME -e EFFECT {-a ACTION ...} {-c AMAZON RESOURCE NAME ...} [-o]
```

Options

Name	Description	Required
`-r` *`ROLENAME`*	Name of the role the policy is for. Type: String Default: None	Yes
`-p` *`POLICYNAME`*	Name you want to assign the policy. Type: String Default: None	Yes
`-e` *`EFFECT`*	The value for the policy's `Effect` element. Specifies whether the policy results in an *allow* or a *deny*. Type: String Valid Values: `Allow` | `Deny` Default: None	Yes
`-a` *`ACTION`*	The value for the policy's `Action` element. Specifies the service and action you want to allow or deny permission to. For example: `-a s3:ListBuckets`. You can use wildcards, and you can specify more than one `-a Action` option in the request. The following example specifies all the Amazon S3 actions related to buckets: `-a s3:*Bucket*` Type: String Default: None	Yes

Name	Description	Required
`-c` *`AMAZON RESOURCE NAME`*	The value for the policy's `Resource` element. Specifies the Amazon Resource Name (ARN) for the resource (or resources) the policy applies to. You can use wildcards, and you can specify more than one `-c AMAZON RESOURCE NAME` option in the request. The following example specifies all the resources in the AWS account: `-c "*"` (quotation marks are required if you're just specifying * as the resource). Type: String Default: None	Yes
`-o`	Causes the output to include the JSON policy document that IAM created for you.	No

Output

If the command is successful, the output is empty. If you specified the `-o` option, the output includes the JSON policy document.

Example

The following example request adds (or updates) the policy named s3access for the role named *myrole*. The `-o` option causes the output to include the JSON policy document we construct for you based on the options you provided.

```
PROMPT> iam-roleaddpolicy -r myrole -p s3access -e Allow -a "s3:*" -c "*" -o

{"Version":"2008-10-17","Statement":[{"Effect":"Allow","Action":["s3:*"],"Re
source":["*"]}]}
```

Related Commands

- iam-rolecreate (p. 48)
- iam-roledel (p. 50)
- iam-roledelpolicy (p. 52)
- iam-rolegetattributes (p. 53)
- iam-rolelistbypath (p. 54)
- iam-rolelistpolicies (p. 56)
- iam-roleupdateassumepolicy (p. 58)
- iam-roleuploadpolicy (p. 60)

iam-rolecreate

Description

Creates a new role in your AWS account.

An AWS account can have only a limited number of roles. For more information, see Appendix A: Limitations on IAM Entities (p. 131).

Syntax

```
iam-rolecreate -r ROLENAME [ -f POLICYDOCUMENTFILE | -s SERVICE] [-p PATH] [-v]
```

Options

Name	Description	Required
`-r` *`ROLENAME`*	Name of the role to create. Do not include the path in this value. Type: String Constraints: See Appendix A: Limitations on IAM Entities (p. 131) Default: None	Yes
`-p` *`PATH`*	Path to the user. For more information about paths, go to Identifiers for IAM Entities in *Using AWS Identity and Access Management*. If you don't want the role to have a path, set to `/`. Type: String Constraints: See Appendix A: Limitations on IAM Entities (p. 131) Default: /	No
`-s` *`SERVICE`*	The entity that can assume the role. Currently, the only acceptable value is the endpoint for Amazon EC2, `ec2.amazonaws.com`. For more information about service endpoints, go to Regions and Endpoints in the *AWS General Reference*. Type: String Condition: Either `-s SERVICE` or `-f POLICYDOCUMENTFILE` is required. If you use both options together, IAM returns an error. Default: None	Conditional

Name	Description	Required
`-f`	Path and name of the file containing the policy. Type: String Condition: Either `-f POLICYDOCUMENTFILE` or `-s SERVICE` is required. If you use both options together, IAM returns an error. Default: None	Conditional
`-v`	Causes the response to include the newly created role's ARN, GUID, and JSON policy document. For more information about ARNs and GUIDs, go to Identifiers for IAM Entities in *Using AWS Identity and Access Management*. Type: String Default: None	No

Output

If the command is successful, the output is empty.

Examples

The following example creates a new role called *myrole* with no path. You could omit the `-p` option and get the same result. The `-v` option causes the output to include the role's ARN, GUID, and JSON policy document.

```
PROMPT> iam-rolecreate -r myrole -p / -s ec2.amazonaws.com -v
arn:aws:iam::123456789012:role/myrole
AROAIFMQYG233LEXAMPLE
{"Version":"2008-10-17","Statement":[{"Effect":"Allow","Principal":{"Ser
vice":["ec2.amazonaws.com"]},"Action":["sts:AssumeRole"]}]}
```

Related Commands

- iam-roleaddpolicy (p. 46)
- iam-roledel (p. 50)
- iam-roledelpolicy (p. 52)
- iam-rolegetattributes (p. 53)
- iam-rolelistbypath (p. 54)
- iam-rolelistpolicies (p. 56)
- iam-roleupdateassumepolicy (p. 58)
- iam-roleuploadpolicy (p. 60)

iam-roledel

Description

Deletes a role from your AWS account. You must remove any attached policies from the role before you can delete it.

To delete a role recursively, use the `-c` option. Recursively deleting the role automatically deletes the associated instance profile and role policies.

Important

Use the `-c` option with caution. Before performing a recursive delete, to ensure you are not deleting anything you don't want to, use the `-p` option along with the `-c` option to list all the role's associated instance profiles without actually performing the recursive deletion.

Caution

Make sure you do not have any Amazon EC2 instances running with the role or instance profile you are about to delete. Deleting a role or instance profile that is associated with a running instance will break any applications running on the instance.

Syntax

```
iam-userdel -r ROLENAME [ -c [-p] ]
```

Options

Name	Description	Required
`-r` *`ROLENAME`*	Name of the role to delete. Type: String Default: None	Yes
`-c`	Deletes the role and deletes any associated instance profiles along with the role. Type: String Default: None	Optional
`-c -p`	Returns what would be deleted, without actually recursively deleting the role. Use this before using `-c` to ensure you are not deleting anything you don't want to. The `-p` option indicates *pretend mode*; use only with the `-c` option. Type: String Constraints: You can apply `-p` only together with `-c`. Default: None	Optional

Output

If the command is successful, the output is empty.

Examples

The following example deletes the role called *myrole*. This example assumes *myrole* has no associated instance profiles.

```
PROMPT> iam-roledel -r myrole
```

The following example shows what would be deleted if you were to recursively delete *myrole*.

```
PROMPT> iam-roledel -r myrole -c -p
instance profiles
     arn:aws:iam::123456789012:instance-profile/myinstanceprofile
```

The following example recursively deletes the role *myrole*.

```
PROMPT> iam-roledel -r myrole -c
```

Related Commands

- iam-roleaddpolicy (p. 46)
- iam-rolecreate (p. 48)
- iam-roledelpolicy (p. 52)
- iam-rolegetattributes (p. 53)
- iam-rolelistbypath (p. 54)
- iam-rolelistpolicies (p. 56)
- iam-roleupdateassumepolicy (p. 58)
- iam-roleuploadpolicy (p. 60)

iam-roledelpolicy

Description

Removes a policy from the specified role.

Syntax

```
iam-roledelpolicy -r ROLENAME -p POLICYNAME
```

Options

Name	Description	Required
`-r` *`ROLENAME`*	Name of the role the policy is attached to. Type: String Default: None	Yes
`-p` *`POLICYNAME`*	Name of the policy document to delete. Type: String Default: None	Yes

Output

If the command is successful, the output is empty.

Example

The following example request deletes the policy named rolepolicy from the role named *myrole*.

```
PROMPT> iam-roledelpolicy -r myrole -p rolepolicy
```

Related Commands

- iam-roleaddpolicy (p. 46)
- iam-rolecreate (p. 48)
- iam-roledel (p. 50)
- iam-rolegetattributes (p. 53)
- iam-rolelistbypath (p. 54)
- iam-rolelistpolicies (p. 56)
- iam-roleupdateassumepolicy (p. 58)
- iam-roleuploadpolicy (p. 60)

iam-rolegetattributes

Description

Returns the Amazon Resource Name (ARN), the GUID, and the assume policy for the specified role. For more information about ARNs, go to ARNs in *Using AWS Identity and Access Management*.

Syntax

```
iam-rolegetattributes -r ROLENAME
```

Options

Name	Description	Required
`-r` *`ROLENAME`*	Name of the role you want to get information about. Type: String Default: None	Yes

Output

The output lists the role's Amazon Resource Name (ARN), GUID, and the associated assume policy.

Examples

The following example returns output for a role named *myrole*. The first line is the ARN, the second line is the role GUID, and the last line is the policy that grants permission to Amazon EC2 to assume the role.

```
PROMPT> iam-rolegetattributes -r myrole
arn:aws:iam::123456789012:role/myrole
AROAJYXWUP72XVEXAMPLE
{"Version":"2008-10-17","Statement":[{"Effect":"Allow","Principal":{"Ser
vice":["ec2.amazonaws.com"]},"Action":["sts:AssumeRole"]}]}
```

Related Commands

- iam-roleaddpolicy (p. 46)
- iam-rolecreate (p. 48)
- iam-roledel (p. 50)
- iam-roledelpolicy (p. 52)
- iam-rolelistbypath (p. 54)
- iam-rolelistpolicies (p. 56)
- iam-roleupdateassumepolicy (p. 58)
- iam-roleuploadpolicy (p. 60)

iam-rolelistbypath

Description

Lists the roles that have the specified path prefix, or lists all the roles in the AWS account. If none exist, the action still succeeds. You can paginate the results using the `MaxItems` and `Marker` options.

Syntax

```
iam-rolelistbypath [-p PATHPREFIX]
```

Options

Name	Description	Required
`-p` *`PATHPREFIX`*	The path prefix for filtering the results. For example, `/division_abc/subdivision_xyz/` would get all roles whose path starts with /division_abc/subdivision_xyz/. Type: String Default: / Condition: Use this option only if you want to list the roles with a specific path prefix.	Conditional
`-i` *`MAXITEMS`*	Use this option only when paginating results to indicate the maximum number of items you want in the response. If there are additional items beyond the maximum you specify, the `IsTruncated` response element is `true`. Type: String Default: None	No
`-m` *`MARKER`*	Use this only when paginating results, and only in a subsequent request after you've received a response where the results are truncated. Set it to the value of the Marker element in the response you just received. Type: String Default: None	No

Output

The output lists the Amazon Resource Name (ARN) for each resulting role. For more information about ARNs, go to ARNs in *Using AWS Identity and Access Management*.

Examples

The following example lists all the roles in the AWS account.

```
PROMPT> iam-rolelistbypath
arn:aws:iam::123456789012:role/myrole
arn:aws:iam::123456789012:role/division_abc/subdivision_xyz/myrole1
arn:aws:iam::123456789012:role/myrole2
IsTruncated: false
```

The following example lists all the roles whose path starts with /division_abc/subdivision_xyz/.

```
PROMPT> iam-rolelistbypath -p /division_abc/subdivision_xyz/
arn:aws:iam::123456789012:role/division_abc/subdivision_xyz/myrole1
IsTruncated: false
```

Related Commands

- iam-roleaddpolicy (p. 46)
- iam-rolecreate (p. 48)
- iam-roledel (p. 50)
- iam-roledelpolicy (p. 52)
- iam-rolegetattributes (p. 53)
- iam-rolelistpolicies (p. 56)
- iam-roleupdateassumepolicy (p. 58)
- iam-roleuploadpolicy (p. 60)

iam-rolelistpolicies

Description

Lists one specific policy or all the policies attached to the specified role. If no policies are attached to the role, the action still succeeds. You can paginate the results using the `MaxItems` and `Marker` options.

Syntax

```
iam-rolelistpolicies -r ROLENAME [-p POLICYNAME] [-v]
```

Options

Name	Description	Required
`-r` *ROLENAME*	Name of the role the policy is attached to. Type: String Default: None	Yes
`-p` *POLICYNAME*	Name of the policy document to display. Type: String Default: None	No
`-v`	Displays the contents of the resulting policies (in addition to the policy names). Type: String	No
`-i` *MAXITEMS*	Use this option only when paginating results to indicate the maximum number of items you want in the response. If there are additional items beyond the maximum you specify, the `IsTruncated` response element is `true`. Type: String Default: None	No
`-m` *MARKER*	Use this only when paginating results, and only in a subsequent request after you've received a response where the results are truncated. Set it to the value of the Marker element in the response you just received. Type: String Default: None	No

Output

The output contains the contents of the specific policy you requested, or it contains the names of the policies attached to the specified role (and optionally the contents of each).

Examples

The following example lists the policy named mypolicy, which is attached to the role named *myrole*. The policy grants permissions to the entity that assumes the role.

```
PROMPT> iam-rolelistpolicies -r myrole -p mypolicy
{"Version":"2008-10-17","Statement":[{"Effect":"Allow","Action":["s3:*"],"Re
source":["*"]}]}
```

The following example lists all policies attached to the role, including the name and the contents of the policies. In this case there is one policy named mypolicy attached to the role.

```
PROMPT> iam-rolelistpolicies -r myrole -v
mypolicy
{"Version":"2008-10-17","Statement":[{"Effect":"Allow","Action":["s3:*"],"Re
source":["*"]}]}
IsTruncated: false
```

Related Commands

- iam-roleaddpolicy (p. 46)
- iam-rolecreate (p. 48)
- iam-roledel (p. 50)
- iam-roledelpolicy (p. 52)
- iam-rolegetattributes (p. 53)
- iam-rolelistbypath (p. 54)
- iam-roleupdateassumepolicy (p. 58)
- iam-roleuploadpolicy (p. 60)

iam-roleupdateassumepolicy

Description

Updates the policy that controls who can assume a given role.

Note

Currently, permission to assume a role is limited to Amazon EC2 instances in your AWS account only.

Syntax

```
iam-roleupdateassumepolicy -r ROLENAME [ -f POLICYDOCUMENTFILE | -s SERVICE]
[-o]
```

Options

Name	Description	Required
`-r` *`ROLENAME`*	Name of the role to update. Type: String Default: None	Yes
`-s` *`SERVICE`*	The entity that can assume the role. Currently, the only entity that can assume a role is an Amazon EC2 instance. For Amazon EC2, the value is the EC2 endpoint `ec2.amazonaws.com`. For more information about service endpoints, go to Regions and Endpoints in the *AWS General Reference*. Type: String Condition: Either `-s SERVICE` or `-f POLICYDOCUMENTFILE` is required. If you use both options together, IAM returns an error. Default: None	Conditional
`-f`	Path and name of the file containing the policy. Type: String Condition: Either `-f POLICYDOCUMENTFILE` or `-s SERVICE` is required. If you use both options together, IAM returns an error. Default: None	Conditional

Name	Description	Required
`-o`	Causes the response to include the new JSON policy document. Type: String Default: None	No

Output

If the command is successful, the output is empty.

Examples

The following example updates a role so that an Amazon EC2 instance can assume the role. The `-o` option causes the output to include the role's JSON policy document.

```
PROMPT> iam-roleupdateassumepolicy -r myrole -s ec2.amazonaws.com -o
{"Version":"2008-10-17","Statement":[{"Effect":"Allow","Principal":{"Ser
vice":["ec2.amazonaws.com"]},"Action":["sts:AssumeRole"]}]}
```

Related Commands

- iam-roleaddpolicy (p. 46)
- iam-rolecreate (p. 48)
- iam-roledel (p. 50)
- iam-roledelpolicy (p. 52)
- iam-rolegetattributes (p. 53)
- iam-rolelistbypath (p. 54)
- iam-rolelistpolicies (p. 56)
- iam-roleuploadpolicy (p. 60)

iam-roleuploadpolicy

Description

Takes a policy you've written and attaches it to the specified role. If a policy with that name is already attached to the role, it's overwritten with the new one. The command accepts either a string representing the policy, or a file containing the policy. For information about the contents of policies, refer to *Using AWS Identity and Access Management*.

Syntax

```
iam-roleuploadpolicy -r ROLENAME -p POLICYNAME [-f POLICYDOCUMENTFILE | -o
POLICYDOCUMENT]
```

Options

Name	Description	Required
-r *ROLENAME*	Name of the role the policy is for. Type: String Default: None	Yes
-p *POLICYNAME*	Name you want to assign the policy. Type: String Default: None	Yes
-f *POLICYDOCUMENTFILE*	Path and name of the file containing the policy. Type: String Condition: Either `-f POLICYDOCUMENTFILE` or `-o POLICYDOCUMENT` is required, but not both. Default: None	Conditional
-o *POLICYDOCUMENT*	The policy (a JSON text string). Type: String Condition: Either `-f POLICYDOCUMENTFILE` or `-o POLICYDOCUMENT` is required, but not both. Default: None	Conditional

Output

If the command is successful, the output is empty.

Example

The following example adds (or updates) the policy named mypolicy for the role named *myrole*. The policy is uploaded as a text string.

```
PROMPT> iam-roleuploadpolicy -r myrole -p mypolicy -o {"Statement":[{"Effect":
 "Allow","Action": "s3:*","Resource":"*"}]}
```

The following example adds (or updates) the policy named mypolicy1 for the role named *myrole1*. The policy is uploaded as a text file.

```
PROMPT> iam-roleuploadpolicy -r myrole1 -p mypolicy1 -f C:\Policies\mypo
licy1_file.txt
```

Related Commands

- iam-roleaddpolicy (p. 46)
- iam-rolecreate (p. 48)
- iam-roledel (p. 50)
- iam-roledelpolicy (p. 52)
- iam-rolegetattributes (p. 53)
- iam-rolelistbypath (p. 54)
- iam-rolelistpolicies (p. 56)
- iam-roleupdateassumepolicy (p. 58)

iam-useraddcert

Description

Some AWS products allow (or require) an X.509 certificate (i.e., a *signing certificate*) and corresponding private key. This command uploads a signing certificate for a user (you must create your own certificate with a tool such as OpenSSL). By default, the certificate's status is `Active` when it's uploaded.

The command accepts either a string representing the contents of the certificate, or the `.pem` certificate file itself.

⚠ **Important**

Calling this command without specifying a user name adds a certificate for the user who owns the requesting credentials.

⚠ **Important**

For Windows users: Due to limitations of the Windows shell, you must upload the certificate as a `.pem` file and must not specify the certificate as a text string.

Syntax

```
iam-useraddcert [-c CERTIFICATE | -f CERTIFICATEFILE] [-u USERNAME]
```

Options

Name	Description	Required
`-c` *`CERTIFICATE`*	The contents of the signing certificate (including the `-----BEGIN CERTIFICATE-----` and `-----END CERTIFICATE-----` lines), and enclosed in quote marks ("). Type: String Condition: Either `-c CERTIFICATE` or `-f CERTIFICATEFILE` is required. If you use both options together, IAM returns an error. Default: None	Conditional
`-f` *`CERTIFICATEFILE`*	The path and name of the `.pem` certificate file. Type: String Condition: Either `-c CERTIFICATE` or `-f CERTIFICATEFILE` is required. If you use both options together, IAM returns an error. Default: None	Conditional

Name	Description	Required
`-u` *`USERNAME`*	Name of the user the signing certificate is for. Type: String Default: None	Optional

Output

The command returns a certificate ID, which you need in order to modify or delete the certificate in the future.

Example

The following example uploads a signing certificate as text for the user named Bob.

```
PROMPT> iam-useraddcert -u Bob -c "-----BEGIN CERTIFICATE-----
    MIICdzCCAeCgAwIBAgIGANc+Ha2wMA0GCSqGSIb3DQEBBQUAMFMxCzAJBgNVBAYT
    AlVTMRMwEQYDVQQKEwpBbWF6b24uY29tMQwwCgYDVQQLEwNBV1MxITAfBgNVBAMT
    GEFXUyBMaW1pdGVkLUFzc3VyYW5jZSBDQTAeFw0wOTAyMDQxNzE5MjdaFw0xMDAy
    MDQxNzE5MjdaMFIxCzAJBgNVBAYTAlVTMRMwEQYDVQQKEwpBbWF6b24uY29tMRcw
    FQYDVQQLEw5BV1MtRGV2ZWxvcGVyczEVMBMGA1UEAxMMNTdxNDl0c3ZwYjRtMIGf
    MA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCpB/vsOwmT/O0td1RqzKjttSBaPjbr
    dqwNe9BrOyB08fw2+Ch5oonZYXfGUrT6mkYXH5fQot9HvASrzAKHO596FdJA6DmL
    ywdWe1Oggk7zFSXO1Xv+3vPrJtaYxYo3eRIp7w80PMkiOv6M0XK8ubcTouODeJbf
    suDqcLnLDxwsvwIDAQABo1cwVTAOBgNVHQ8BAf8EBAMCBaAwFgYDVR0lAQH/BAww
    CgYIKwYBBQUHAwIwDAYDVR0TAQH/BAIwADAdBgNVHQ4EFgQULGNaBphBumaKbDRK
    CAi0mH8B3mowDQYJKoZIhvcNAQEFBQADgYEAuKxhkXaCLGcqDuweKtO/AEw9ZePH
    wr0XqsaIK2HZboqruebXEGsojK4Ks0WzwgrEynuHJwTn760xe39rSqXWIOGrOBaX
    wFpWHVjTFMKk+tSDG1lssLHyYWWdFFU4AnejRGORJYNaRHgVTKjHphc5jEhHm0BX
    AEaHzTpmEXAMPLE=
    -----END CERTIFICATE-----"
TA7SMP42TDN5Z26OBPJE7EXAMPLE
```

The following example uploads a signing certificate as a `.pem` file for the user named Bob.

```
PROMPT> iam-useraddcert -u Bob -f C:\Certs\Bob_Cert_1.pem
TA7SMP42TDN5Z26OBPJE7EXAMPLE
```

Related Commands

- iam-userdelcert (p. 76)
- iam-userlistcerts (p. 87)
- iam-usermodcert (p. 99)

iam-useraddkey

Description

Creates a new AWS Secret Access Key and corresponding AWS Access Key ID for the specified user. When the key is created, its status is `Active` by default. A user can have only a limited number of keys. For more information, see Appendix A: Limitations on IAM Entities (p. 131).

Important

Calling this command without specifying a user name creates keys for the user who owns the requesting credentials.

Important

To ensure the security of your AWS account, the secret access key is accessible only during key and user creation. You must save the key (for example, in a text file) if you want to be able to access it again. If a secret key is lost, you can delete the access keys for the associated user and then create new keys. For more information about deleting keys, see iam-userdelkey (p. 77).

Syntax

```
iam-useraddkey [-u USERNAME]
```

Options

Name	Description	Required
`-u` *USERNAME*	The user the new key will belong to. Type: String Default: None	Optional

Output

The output includes two lines: The first one lists the user's new Access Key ID, and the second lists the corresponding new Secret Access Key.

Examples

The following example creates a new access key for the user Bob.

```
PROMPT> iam-useraddkey -u Bob
AKIAIOSFODNN7EXAMPLE
wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
```

Related Commands

- iam-userdelkey (p. 77)
- iam-userlistkeys (p. 91)
- iam-usermodkey (p. 101)

iam-useraddloginprofile

Description

Creates a password for the specified user.

Syntax

```
iam-useraddloginprofile -u USERNAME -p PASSWORD
```

Options

Name	Description	Required
`-u` *USERNAME*	Name of the user the password is assigned to. Type: String Default: None	Yes
`-p` *PASSWORD*	The password for the user. Type: String Default: None	Yes

Output

If the command is successful, the output is empty.

Example

The following example creates a the password *Welcome* for a user called Bob.

```
PROMPT> iam-useraddloginprofile -u Bob -p Welcome
```

Related Commands

- iam-usercreate (p. 71)
- iam-userdelloginprofile (p. 78)
- iam-usergetloginprofile (p. 84)
- iam-usermodloginprofile (p. 103)

iam-useraddpolicy

Description

Creates a policy based on the information you provide and attaches the policy to the specified user. Use this command if you need a simple policy with no conditions, and you don't want to write the policy yourself. If you need a policy with conditions, you must write the policy yourself and upload it with iam-useruploadpolicy (p. 106). For information about the contents of policies, refer to *Using AWS Identity and Access Management*.

A user can have only a limited number of policies. For more information, see Appendix A: Limitations on IAM Entities (p. 131).

Syntax

```
iam-useraddpolicy -u USERNAME -p POLICYNAME -e EFFECT {-a ACTION ...} {-r AMAZON
RESOURCE NAME ...} [-o]
```

Options

Name	Description	Required
`-u` *`USERNAME`*	Name of the user the policy is for. Type: String Default: None	Yes
`-p` *`POLICYNAME`*	Name you want to assign the policy. Type: String Default: None	Yes
`-e` *`EFFECT`*	The value for the policy's `Effect` element. Specifies whether the policy results in an *allow* or a *deny*. For more information about policies and their contents, refer to *Using AWS Identity and Access Management*. Type: String Valid Values: `Allow` | `Deny` Default: None	Yes

Name	Description	Required
`-a` *`ACTION`*	The value for the policy's `Action` element. Specifies the service and action you want to allow or deny permission to. For example: `-a iam:ListAccessKeys`. You can use wildcards, and you can specify more than one `-a Action` option in the request. The following example specifies all the IAM actions related to access keys or signing certificates: `-a iam:*AccessKey* -a iam:*SigningCertificate*` Type: String Default: None	Yes
`-r` *`AMAZON RESOURCE NAME`*	The value for the policy's `Resource` element. Specifies the Amazon Resource Name (ARN) for the resource (or resources) the policy applies to. You can use wildcards, and you can specify more than one `-r AMAZON RESOURCE NAME` option in the request. The following example specifies all the resources in the AWS account: `-r "*"` (quotation marks are required if you're just specifying * as the resource). The following example specifies all groups in the AWS account: `-r arn:aws:iam::123456789012:group/*` Type: String Default: None	Yes
`-o`	Causes the output to include the JSON policy document that IAM created for you.	No

Output

If the command is successful, the output is empty. Exception: if you specified the `-o` option, the output includes the JSON policy document.

Example

The following example request adds (or updates) the policy named AdminRoot for the user named Bob. The `-o` option causes the output to include the JSON policy document we construct for you based on the options you provided.

```
PROMPT> iam-useraddpolicy -u Bob -p AdminRoot -e Allow -a "*" -r "*" -o

{"Version":"2008-10-17","Statement":[{"Effect":"Allow","Action":["*"],"Re
source":["*"]}]}
```

Related Commands

- iam-userdelpolicy (p. 79)

- iam-userlistpolicies (p. 95)
- iam-useruploadpolicy (p. 106)

iam-userchangepassword

Description

Changes the password of the IAM user calling `iam-userchangepassword`. The root account password is not affected by this command.

For information about changing passwords, go to Managing Passwords in *Using IAM*.

Syntax

```
iam-userchangepassword -o OLDPASSWORD -n NEWPASSWORD
```

Options

Name	Description	Required
`-o` *`OLDPASSWORD`*	Your current password. Type: String Default: None	Yes
`-p` *`NEWPASSWORD`*	Your new password. Type: String Constraints: Any constraints imposed by a password policy on the account. For information about password policies, go to Managing an IAM Password Policy in *Using AWS Identity and Access Management*. For information about limitations on passwords, see Appendix A: Limitations on IAM Entities (p. 131). Default: None	Yes

Output

If the command is successful, the output is empty.

Examples

The following example changes the old password, Welcome1, to the new password, Welcome2.

```
PROMPT> iam-userchangepassword -o Welcome1 -n Welcome2
```

Related Commands

- iam-useraddloginprofile (p. 66)
- iam-userdelloginprofile (p. 78)

iam-usercreate

Description

Creates a new user in your AWS account. Optionally adds the user to one or more groups, and creates an access key for the user.

An AWS account can have only a limited number of users. For more information, see Appendix A: Limitations on IAM Entities (p. 131).

Important

To ensure the security of your AWS account, the secret access key is accessible only during key and user creation. You must save the key (for example, in a text file) if you want to be able to access it again. If a secret key is lost, you can delete the access keys for the associated user and then create new keys. For more information about deleting keys and adding keys, see iam-userdelkey (p. 77) and iam-useraddkey (p. 64)

Syntax

```
iam-usercreate -u USERNAME [-p PATH] [-g GROUPS ...] [-k] [-v]
```

Options

Name	Description	Required
`-u` *`USERNAME`*	Name of the user to create. Do not include the path in this value. Type: String Constraints: See Appendix A: Limitations on IAM Entities (p. 131) Default: None	Yes
`-p` *`PATH`*	Path to the user. For more information about paths, go to Identifiers for IAM Entities in *Using AWS Identity and Access Management*. If you don't want the user to have a path, set to `/`. Type: String Constraints: See Appendix A: Limitations on IAM Entities (p. 131) Default: /	No
`-g` *`GROUPS`*	Name of a group you want to add the user to. Repeat this option for each group you want to add the user to. Type: String Default: None	No

Name	Description	Required
`-k`	Creates an access key for the user. Type: String Default: None	No
`-v`	Causes the response to include the newly created user's ARN and GUID. For more information about ARNs and GUIDs, go to Identifiers for IAM Entities in *Using AWS Identity and Access Management*. Type: String Default: None	No

Output

The output is empty unless you requested to create an access key for the user. In that case, the output includes the Access Key ID and Secret Access Key.

Examples

The following example creates a new user called Bob with no path. You could omit the `-p` option and get the same result. The `-v` option causes the output to include the user's ARN and GUID.

```
PROMPT> iam-usercreate -u Bob -p / -v
arn:aws:iam::123456789012:user/division_abc/subdivision_xyz/Bob
AIDXYZ7REHA4ODEXAMPLE
```

The following example creates a new user called Bob with a path of /division_abc/subdivision_xyz/product_1234/.

```
PROMPT> iam-usercreate -u Bob -p /division_abc/subdivision_xyz/product_1234/
```

The following example creates a new user called Bob, adds the user to the Developers group and the AllUsers group, and creates an access key for the user.

```
PROMPT> iam-usercreate -u Bob -g Developers -g AllUsers -k
AKIAIOSFODNN7EXAMPLE
wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
```

Related Commands

- iam-userdel (p. 74)
- iam-userlistbypath (p. 85)
- iam-usermod (p. 97)

iam-userdeactivatemfadevice

Description

Deactivates the specified MFA device associated with the specified user.

Syntax

```
iam-userdeactivatemfadevice -u USERNAME -s SERIAL
```

Options

Name	Description	Required
`-u` *`USERNAME`*	Name of the user you want to list the MFA device for. Type: String Default: None	Yes
`-s` *`SERIAL`*	The serial number for the MFA device to deactivate. Type: String Default: None	Yes

Output

If the command is successful, the output is empty.

Example

The following example deactivates the MFA device with a serial number GATTxxxx32X for a user called John.

```
PROMPT> iam-userdeactivatemfadevice -u John -s GATTxxxx32X
```

Related Commands

- iam-userenablemfadevice (p. 80)
- iam-userresyncmfadevice (p. 104)
- iam-userlistmfadevices (p. 93)
- iam-virtualmfadevicecreate (p. 125)
- iam-virtualmfadevicedel (p. 127)

iam-userdel

Description

Deletes a user from your AWS account. When this command is used only with the `-u` option, the user must not belong to any groups, have any keys or signing certificates, or have any attached policies.

To delete the user recursively, use the `-r` option. Recursively deleting the user automatically deletes it from any associated groups and deletes any attached entities such as keys, signing certificates, and policies.

⚠ **Important**

Use the `-r` option with caution. Before performing a recursive delete, to ensure you are not deleting anything you don't want to, use the `-p` option along with the `-r` option to list all the user's associated entities and groups without actually performing the recursive deletion.

Syntax

```
iam-userdel -u USERNAME [ -r [-p] ]
```

Options

Name	Description	Required
`-u` *USERNAME*	Name of the user to delete. Type: String Default: None	Yes
`-r`	Deletes the user from associated groups and deletes the user's credentials and policies along with the user. Type: String Default: None	Optional
`-r -p`	Returns what would be deleted, without actually recursively deleting the user. Use this before using `-r` to ensure you are not deleting anything you don't want to. The `-p` option indicates *pretend mode*; use only with the `-r` option. Type: String Constraints: You can apply -p only together with -r. Default: None	Optional

Output

If the command is successful, the output is empty.

Examples

The following example deletes the user called Jack. This example assumes Jack is in no groups and has no policies.

```
PROMPT> iam-userdel -u Jack
```

The following example shows what would be deleted if you were to recursively delete the user Jack.

```
PROMPT> iam-userdel -u Jack -r -p
accesskeys
     AKIAIOSFODNN7EXAMPLE
policies
     KeyPolicy
certificates
     TA7SMP42TDN5Z26OBPJE7EXAMPLE
groups
     arn:aws:iam::123456789012:group/Managers
     arn:aws:iam::123456789012:group/Finance
```

The following example recursively deletes user Jack.

```
PROMPT> iam-userdel -u Jack -r
```

Related Commands

- iam-usercreate (p. 71)
- iam-userlistbypath (p. 85)
- iam-usermod (p. 97)

iam-userdelcert

Description

Deletes the specified signing certificate belonging to the specified user.

> **Important**
>
> Calling this command without specifying a user name deletes the certificate for the user who owns the requesting credentials.

Syntax

```
iam-userdelcert -c CERTIFICATEID [-u USERNAME]
```

Options

Name	Description	Required
-c *CERTIFICATEID*	ID of the signing certificate to delete. Type: String Default: None	Yes
-u *USERNAME*	Name of the user the signing certificate belongs to. Type: String Default: None	Optional

Output

If the command is successful, the output is empty.

Examples

The following example deletes the signing certificate with ID TA7SMP42TDN5Z26OBPJE7EXAMPLE, which belongs to the user named Bob.

```
PROMPT> iam-userdelcert -c TA7SMP42TDN5Z26OBPJE7EXAMPLE -u Bob
```

Related Commands

- iam-useraddcert (p. 62)
- iam-userlistcerts (p. 87)
- iam-usermodcert (p. 99)

iam-userdelkey

Description

Deletes the specified Access Key ID and corresponding Secret Access Key for the specified user.

Important

Calling this command without specifying a user name deletes the keys for the user who owns the requesting credentials.

Syntax

```
iam-userdelkey [-u USERNAME] -k ACCESSKEYID
```

Options

Name	Description	Required
-u *USERNAME*	Name of the user whose key you want to delete. Type: String Default: None	Optional
-k *ACCESSKEYID*	The Access Key ID for the Secret Access Key you want to delete. Type: String Default: None	Yes

Output

If the key is successfully deleted, the output is empty.

Examples

The following example deletes the Secret Access Key with Access Key ID AKIAIOSFODNN7EXAMPLE from the user named Bob.

```
PROMPT> iam-userdelkey -u Bob -k AKIAIOSFODNN7EXAMPLE
```

Related Commands

- iam-useraddkey (p. 64)
- iam-userlistkeys (p. 91)
- iam-usermodkey (p. 101)

iam-userdelloginprofile

Description

Removes the password for the specified user. Removing the password disables a user's ability to sign in to AWS through the AWS Management Console.

Syntax

```
iam-userdelloginprofile -u USERNAME
```

Options

Name	Description	Required
`-u` *`USERNAME`*	Name of the user whose password you want to delete. Type: String Default: None	Yes

Output

If the command is successful, the output is empty.

Example

The following example deletes a password for a user called Susan.

```
PROMPT> iam-userdelloginprofile -u Susan
```

Related Commands

- iam-userdel (p. 74)
- iam-useraddloginprofile (p. 66)
- iam-usergetloginprofile (p. 84)
- iam-usermodloginprofile (p. 103)

iam-userdelpolicy

Description

Removes a policy from the specified user.

Syntax

```
iam-userdelpolicy -u USERNAME -p POLICYNAME
```

Options

Name	Description	Required
`-u` *USERNAME*	Name of the user the policy is attached to. Type: String Default: None	Yes
`-p` *POLICYNAME*	Name of the policy document to delete. Type: String Default: None	Yes

Output

If the command is successful, the output is empty.

Example

The following example request deletes the policy named KeyPolicy from the user named Bob.

```
PROMPT> iam-userdelpolicy -u Bob -p KeyPolicy
```

Related Commands

- iam-useraddpolicy (p. 67)
- iam-userlistpolicies (p. 95)
- iam-useruploadpolicy (p. 106)

iam-userenablemfadevice

Description

Enables an MFA device for a user.

Syntax

```
iam-userenablemfadevice -u USERNAME -s SERIAL -c1 CODE 1 -c2 CODE 2
```

Options

Name	Description	Required
`-u USERNAME`	Name of the user to enable the MFA device for. Type: String Default: None	Yes
`-s SERIAL`	The serial number of the MFA device. Type: String Default: None	Yes
`-c1 CODE 1`	The first authentication code for the MFA device. Type: String Default: None	Yes
`-c2 CODE 2`	The second authentication code for the MFA device. Type: String Default: None	Yes

Output

If the command is successful, the output is empty.

Example

The following example enables an MFA device for a user called John, with a serial number GATTxxxx32X and authentication codes of 94xx49, and 97xx97, respectively.

```
PROMPT> iam-userenablemfadevice -u John -s GATTxxxx32X -c1 94xx49 -c2 97xx97
```

Related Commands

- iam-userresyncmfadevice (p. 104)

- iam-userlistmfadevices (p. 93)
- iam-userdeactivatemfadevice (p. 73)
- iam-virtualmfadevicecreate (p. 125)
- iam-virtualmfadevicedel (p. 127)

iam-usergetattributes

Description

Returns Amazon Resource Name (ARN) and user ID of a specified user. When no user is specified, returns the ARN and user ID of the user who owns the requesting credentials.

Syntax

```
iam-usergetattributes [-u USERNAME]
```

Options

Name	Description	Required
`-u` *USERNAME*	Name of the user you want to get information about. Type: String Default: None	Optional

Output

When you use the `-u USERNAME` option to specify a user, the output lists the Amazon Resource Name (ARN) and user ID for that user. When no user is specified, the output lists the ARN and the AWS account ID of the user making the request.

Examples

The following example returns output for a user called Bob. The first line is the ARN and the second line is the user ID.

```
PROMPT> iam-usergetattributes -u Bob
arn:aws:iam::123456789012:user/division_abc/subdivision_xyz/Bob
AIDACKCEVSQ6C2EXAMPLE
```

Note that the above example returns identical output when no user is specified and the requester's AWS account credentials belong to Bob. The first line is the ARN of the user making the call (Bob). The second line is Bob's AWS account ID.

The following example returns output for when no user is specified and the caller owns the root AWS account credentials. The first line is the ARN of the user making the call, with the user name indicated as *root*. The second line is the AWS account ID.

```
PROMPT> iam-usergetattributes
arn:aws:iam::123456789012:root
123456789012
```

Related Commands

- iam-userdel (p. 74)
- iam-userlistbypath (p. 85)
- iam-usermod (p. 97)

iam-usergetloginprofile

Description

Returns verification that a password exists for a specified user, or returns an error if the specified user does not have a password.

Syntax

iam-usergetloginprofile -u *USERNAME*

Options

Name	Description	Required
-u *USERNAME*	Name of the user whose password you want to verify. Type: String Default: None	Yes

Output

The output either returns a message verifying that a password exists for the specified user, or if the user has no password it returns an error message.

Example

The following example returns output when a password exists for a user called John.

```
PROMPT> iam-usergetloginprofile -u John
Login Profile Exists for User John
```

The following example returns output when a password doesn't exist for a user called Jackie.

```
PROMPT> iam-usergetloginprofile -u Jackie
404 NoSuchEntity Cannot find Login Profile for User Jackie
```

Related Commands

- iam-useraddloginprofile (p. 66)
- iam-userdelloginprofile (p. 78)
- iam-usermodloginprofile (p. 103)

iam-userlistbypath

Description

Lists the users that have the specified path prefix, or lists all the users in the AWS account. If none exist, the action still succeeds. You can paginate the results using the `MaxItems` and `Marker` options.

Syntax

```
iam-userlistbypath [-p PATH]
```

Options

Name	Description	Required
`-p` *`PATH`*	The path prefix for filtering the results. For example, `/division_abc/subdivision_xyz/` would get all users whose path starts with /division_abc/subdivision_xyz/. Type: String Default: / Condition: Provide this option only if you want to list the users with a specific path prefix.	Conditional
`-i` *`MAXITEMS`*	Use this option only when paginating results to indicate the maximum number of items you want in the response. If there are additional items beyond the maximum you specify, the `IsTruncated` response element is `true`. Type: String Default: None	No
`-m` *`MARKER`*	Use this only when paginating results, and only in a subsequent request after you've received a response where the results are truncated. Set it to the value of the Marker element in the response you just received. Type: String Default: None	No

Output

The output lists the Amazon Resource Name (ARN) for each resulting user.

Examples

The following example lists all the users in the AWS account.

```
PROMPT> iam-userlistbypath
arn:aws:iam::123456789012:user/division_abc/subdivision_xyz/Bob
arn:aws:iam::123456789012:user/division_abc/subdivision_xyz/Susan
arn:aws:iam::123456789012:user/division_abc/subdivision_xyz/John
arn:aws:iam::123456789012:user/division_abc/subdivision_xyz/engineering/Andrew
arn:aws:iam::123456789012:user/division_abc/subdivision_xyz/engineering/Jackie
```

The following example lists all the users whose path starts with /division_abc/subdivision_xyz/.

```
PROMPT> iam-userlistbypath -p /division_abc/subdivision_xyz/
arn:aws:iam::123456789012:user/division_abc/subdivision_xyz/Bob
arn:aws:iam::123456789012:user/division_abc/subdivision_xyz/Susan
arn:aws:iam::123456789012:user/division_abc/subdivision_xyz/John
```

Related Commands

- iam-usercreate (p. 71)
- iam-userdel (p. 74)

iam-userlistcerts

Description

Lists the signing certificates for a user. If the user has none, the action still succeeds. You can paginate the results using the `MaxItems` and `Marker` options.

Note

Calling this command without specifying a user name lists the certificates for the user who owns the requesting credentials.

Syntax

```
iam-userlistcerts [-u USERNAME] [-v]
```

Options

Name	Description	Required
`-u` *`USERNAME`*	Name of the user. Type: String Default: None	Optional
`-v`	Causes the output to include the signing certificate's contents (in addition to the certificate ID). Type: String	No
`-i` *`MAXITEMS`*	Use this option only when paginating results to indicate the maximum number of items you want in the response. If there are additional items beyond the maximum you specify, the `IsTruncated` response element is `true`. Type: String Default: None	No
`-m` *`MARKER`*	Use this only when paginating results, and only in a subsequent request after you've received a response where the results are truncated. Set it to the value of the Marker element in the response you just received. Type: String Default: None	No

Output

The response lists the certificate ID and the current status of the certificate (`Active` or `Disabled`). Includes the certificate itself if you use the `-v` option.

Examples

The following example lists the signing certificates for the user named Bob.

```
PROMPT> iam-userlistcerts -u Bob -v
TA7SMP42TDN5Z26OBPJE7EXAMPLE
-----BEGIN CERTIFICATE-----
    MIICdzCCAeCgAwIBAgIGANc+Ha2wMA0GCSqGSIb3DQEBBQUAMFMxCzAJBgNVBAYT
    AlVTMRMwEQYDVQQKEwpBbWF6b24uY29tMQwwCgYDVQQLEwNBV1MxITAfBgNVBAMT
    GEFXUyBMaW1pdGVkLUFzc3VyYW5jZSBDQTAeFw0wOTAyMDQxNzE5MjdaFw0xMDAy
    MDQxNzE5MjdaMFIxCzAJBgNVBAYTAlVTMRMwEQYDVQQKEwpBbWF6b24uY29tMRcw
    FQYDVQQLEw5BV1MtRGV2ZWxvcGVyczEVMBMGA1UEAxMMNTdxNDl0c3ZwYjRtMIGf
    MA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCpB/vsOwmT/O0td1RqzKjttSBaPjbr
    dqwNe9BrOyB08fw2+Ch5oonZYXfGUrT6mkYXH5fQot9HvASrzAKHO596FdJA6DmL
    ywdWe1Oggk7zFSXO1Xv+3vPrJtaYxYo3eRIp7w80PMkiOv6M0XK8ubcTouODeJbf
    suDqcLnLDxwsvwIDAQABo1cwVTAOBgNVHQ8BAf8EBAMCBaAwFgYDVR0lAQH/BAww
    CgYIKwYBBQUHAwIwDAYDVR0TAQH/BAIwADAdBgNVHQ4EFgQULGNaBphBumaKbDRK
    CAi0mH8B3mowDQYJKoZIhvcNAQEFBQADgYEAuKxhkXaCLGcqDuweKtO/AEw9ZePH
    wr0XqsaIK2HZboqruebXEGsojK4Ks0WzwgrEynuHJwTn760xe39rSqXWIOGrOBaX
    wFpWHVjTFMKk+tSDG1lssLHyYWWdFFU4AnejRGORJYNaRHgVTKjHphc5jEhHm0BX
    AEaHzTpmEXAMPLE=
    -----END CERTIFICATE-----
Active
```

Related Commands

- iam-useraddcert (p. 62)
- iam-userdelcert (p. 76)
- iam-usermodcert (p. 99)

iam-userlistgroups

Description

Returns the group membership for the specified user. You can paginate the results using the `MaxItems` and `Marker` options.

Syntax

```
iam-userlistgroups -u USERNAME
```

Options

Name	Description	Required
`-u` *`USERNAME`*	The name of the user. Type: String	Yes
`-i` *`MAXITEMS`*	Use this option only when paginating results to indicate the maximum number of items you want in the response. If there are additional items beyond the maximum you specify, the `IsTruncated` response element is `true`. Type: String Default: None	No
`-m` *`MARKER`*	Use this only when paginating results, and only in a subsequent request after you've received a response where the results are truncated. Set it to the value of the Marker element in the response you just received. Type: String Default: None	No

Output

The output lists the Amazon Resource Name (ARN) for each group the user belongs to.

Example

The following example lists the groups the user Bob is in.

```
PROMPT> iam-userlistgroups -u Bob
arn:aws:iam::123456789012:group/division_abc/subdivision_xyz/product_1234/engin
eering/Test
arn:aws:iam::123456789012:group/division_abc/subdivision_xyz/product_1234/Man
agers
```

Related Commands

- iam-groupadduser (p. 15)
- iam-groupremoveuser (p. 29)

iam-userlistkeys

Description

Returns information about the access keys belonging to the specified user. If the user has none, the action still succeeds. You can paginate the results using the `MaxItems` and `Marker` options.

Note

Calling this command without specifying a user name returns information about the access keys for the user who owns the requesting credentials.

Syntax

```
iam-userlistkeys [-u USERNAME]
```

Options

Name	Description	Required
`-u` *`USERNAME`*	Name of the user. Type: String Default: None	Optional
`-i` *`MAXITEMS`*	Use this option only when paginating results to indicate the maximum number of items you want in the response. If there are additional items beyond the maximum you specify, the `IsTruncated` response element is `true`. Type: String Default: None	No
`-m` *`MARKER`*	Use this only when paginating results, and only in a subsequent request after you've received a response where the results are truncated. Set it to the value of the Marker element in the response you just received. Type: String Default: None	No

Output

The response lists the Access Key ID and the current status of the key (`Active` or `Disabled`).

Examples

The following example lists the keys for the user named Bob.

```
PROMPT> iam-userlistkeys -u Bob
AKIAIOSFODNN7EXAMPLE
Active
```

Related Commands

- iam-useraddkey (p. 64)
- iam-userdelkey (p. 77)
- iam-usermodkey (p. 101)

iam-userlistmfadevices

Description

Lists the MFA devices. If the request includes the user name, then this command lists all the MFA device serial numbers associated with the specified user name. If you do not specify a user name, IAM determines the user name implicitly based on the AWS Access Key ID making the request. You can paginate the results using the `MaxItems` and `Marker` options.

Syntax

```
iam-userlistmfadevices -u USERNAME
```

Options

Name	Description	Required
`-u` *`USERNAME`*	Name of the user you want to list the MFA device for. If you omit this option, then IAM returns all MFA devices for the Access Key ID making the request. Type: String Default: None	No
`-i` *`MAXITEMS`*	Use this option only when paginating results to indicate the maximum number of items you want in the response. If there are additional items beyond the maximum you specify, the `IsTruncated` response element is `true`. Type: String Default: None	No
`-m` *`MARKER`*	Use this only when paginating results, and only in a subsequent request after you've received a response where the results are truncated. Set it to the value of the Marker element in the response you just received. Type: String Default: None	No

Output

The command lists the serial numbers of the MFA devices attached to the specified user. If the user has no MFA devices, the output is empty.

Example

The following example lists the MFA devices for a user called Susan.

```
PROMPT> iam-userlistmfadevices -u Susan
   GATTxxxx32X
```

Related Commands

- iam-userenablemfadevice (p. 80)
- iam-userresyncmfadevice (p. 104)
- iam-userdeactivatemfadevice (p. 73)
- iam-virtualmfadevicecreate (p. 125)
- iam-virtualmfadevicedel (p. 127)

iam-userlistpolicies

Description

Lists one specific policy or all the policies attached to the specified user. If no policies are attached to the user, the action still succeeds. You can paginate the results using the `MaxItems` and `Marker` options.

Syntax

```
iam-userlistpolicies -u USERNAME [-p POLICYNAME] [-v]
```

Options

Name	Description	Required
`-u` *`USERNAME`*	Name of the user the policy is attached to. Type: String Default: None	Yes
`-p` *`POLICYNAME`*	Name of the policy document to display. Type: String Default: None	No
`-v`	Displays the contents of the resulting policies (in addition to the policy names). Type: String	No
`-i` *`MAXITEMS`*	Use this option only when paginating results to indicate the maximum number of items you want in the response. If there are additional items beyond the maximum you specify, the `IsTruncated` response element is `true`. Type: String Default: None	No
`-m` *`MARKER`*	Use this only when paginating results, and only in a subsequent request after you've received a response where the results are truncated. Set it to the value of the Marker element in the response you just received. Type: String Default: None	No

Output

The output contains the contents of the specific policy you requested, or it contains the names of the policies attached to the specified user (and optionally the contents of each).

Examples

The following example request displays the policy named KeyPolicy, which is attached to the user named Bob.

```
PROMPT> iam-userlistpolicies -u Bob -p KeyPolicy -v
{"Statement":[{"Effect":"Allow","Action":["iam:*AccessKey*","iam:*SigningCerti
ficate*"],"Resource":"arn:aws:iam::123456789012:user/division_abc/subdivi
sion_xyz/Bob"}]}
```

The following example request displays the names of all the policies attached to the user named Bob. You could optionally have the output display the contents of the policies by including the `-v` option.

```
PROMPT> iam-userlistpolicies -u Bob
KeyPolicy
AnotherBobPolicy
```

Related Commands

- iam-useraddpolicy (p. 67)
- iam-userdelpolicy (p. 79)
- iam-useruploadpolicy (p. 106)

iam-usermod

Description

Changes the user's name or path (or both).

Important

You need to understand the implications of changing a user's path or name. For more information, see Renaming Users and Groups in *Using AWS Identity and Access Management*.

Note

To change a user name the requester must have appropriate permissions on both the source object and the target object. For example, to change Bob to Robert, the entity making the request must have permission on Bob and Robert, or must have permission on all (*). For more information about permissions, see Permissions and Policies.

Syntax

```
iam-usermod -u USERNAME [-n NEWUSERNAME] [-p PATH]
```

Options

Name	Description	Required
`-u` *`USERNAME`*	Name of the user to update. If you're changing the user's name, this is the original name. Type: String Default: None	Yes
`-n` *`NEWUSERNAME`*	New name for the user. Type: String Default: None Condition: Only include this if changing the user's name.	Conditional
`-p` *`PATH`*	New path for the user. Type: String Default: None Condition: Only include this if changing the user's path.	Conditional

Output

If the command is successful, the output is empty.

Examples

The following example changes the name of the user from Bob to Robert.

```
PROMPT> iam-usermod -u Bob -n Robert
```

The following example changes the path for Bob to /division_abc/subdivision_xyz/test/.

```
PROMPT> iam-usermod -u Bob -p /division_abc/subdivision_xyz/test/
```

Related Commands

- iam-usercreate (p. 71)
- iam-userdel (p. 74)
- iam-userlistbypath (p. 85)

iam-usermodcert

Description

Changes the status of the specified signing certificate from active to inactive, or vice versa. This action lets you rotate a user's certificates, or immediately disable (or re enable) a user's ability to make API calls to AWS. For information about rotating certificates, go to Managing Keys and Certificates in *Using AWS Identity and Access Management*.

Syntax

```
iam-usermodcert [-u USERNAME] -c CERTIFICATEID -s Active|Inactive
```

Options

Name	Description	Required
`-u` *`USERNAME`*	Name of the user whose certificate you want to update. Type: String Default: None	Optional
`-c` *`CERTIFICATEID`*	The ID of the signing certificate you want to update. Type: String Default: None	Yes
`-s Active|Inactive`	The status you want to assign to the certificate. `Active` means the user can use the certificate for API calls to AWS. `Inactive` means the user cannot use the certificate for API calls to AWS. Type: String	Yes

Output

If the certificate is successfully updated, the output is empty.

Examples

The following example changes the status to `Inactive` for the certificate ID TA7SMP42TDN5Z26OBPJE7EXAMPLE, which belongs to the user named Bob.

```
PROMPT> iam-usermodcert -u Bob -c TA7SMP42TDN5Z26OBPJE7EXAMPLE -s Inactive
```

Related Commands

- iam-useraddcert (p. 62)
- iam-userdelcert (p. 76)

- iam-userlistcerts (p. 87)

iam-usermodkey

Description

Changes the status of the specified key from active to inactive, or vice versa. This action lets you rotate a user's keys, or immediately disable (or re enable) a user's ability to make API calls to AWS. For information about rotating keys, go to Managing Keys and Certificates in *Using AWS Identity and Access Management*.

⚠ **Important**

Calling this command without specifying a user name modifies the key for the user who owns the requesting credentials.

Syntax

```
iam-usermodkey [-u USERNAME] -k ACCESSKEYID -s Active|Inactive
```

Options

Name	Description	Required
`-u USERNAME`	Name of the user whose key you want to update. Type: String Default: None	Optional
`-k ACCESSKEYID`	The Access Key ID of the Secret Access Key you want to update. Type: String Default: None	Yes
`-s Active|Inactive`	The status you want to assign to the Secret Access Key. `Active` means the user can use the key for API calls to AWS. `Inactive` means the user cannot use the key for API calls to AWS. Type: String	Yes

Output

If the key is successfully updated, the output is empty.

Example

The following example changes the status to `Inactive` for the Secret Access Key with Access Key ID AKIAIOSFODNN7EXAMPLE, which belongs to the user named Bob.

```
PROMPT> iam-usermodkey -u Bob -k AKIAIOSFODNN7EXAMPLE -s Inactive
```

Related Commands

- iam-useraddkey (p. 64)
- iam-userdelkey (p. 77)
- iam-userlistkeys (p. 91)

iam-usermodloginprofile

Description

Changes the password for the specified user. The previous password is overwritten with the new password.

Syntax

```
iam-usergetloginprofile -u USERNAME -p PASSWORD
```

Options

Name	Description	Required
-u *USERNAME*	Name of the user whose password you want to change. Type: String Default: None	Yes
-p *PASSWORD*	The new password for the specified user. Type: String Default: None	Yes

Output

If the command is successful, the output is empty.

Example

The following example changes the password to *Welcome* for a user called Andrew.

```
PROMPT> iam-usermodloginprofile -u Andrew -p Welcome
```

Related Commands

- iam-useraddloginprofile (p. 66)
- iam-userdelloginprofile (p. 78)
- iam-usergetloginprofile (p. 84)

iam-userresyncmfadevice

Description

Re synchronizes an MFA device.

Syntax

```
iam-userresyncmfadevice -u USERNAME -s SERIAL -c1 CODE 1 -c2 CODE 2
```

Options

Name	Description	Required
-u *USERNAME*	Name of the user to re synchronize the MFA device for. Type: String Default: None	Yes
-s *SERIAL*	The serial number of the MFA device. Type: String Default: None	Yes
-c1 *CODE 1*	The first authentication code of the MFA device. Type: String Default: None	Yes
-c2 *CODE 2*	The second authentication code of the MFA device. Type: String Default: None	Yes

Output

If the command is successful, the output is empty.

Example

The following example resynchronizes an MFA device for a user called John, with a serial number GATTxxxx32X, and authentication codes of 94xx49, and 97xx97, respectively.

```
PROMPT> iam-userresyncmfadevice -u John -s GATTxxxx32X -c1 94xx49 -c2 97xx97
```

Related Commands

- iam-userenablemfadevice (p. 80)

- iam-userlistmfadevices (p. 93)
- iam-userdeactivatemfadevice (p. 73)
- iam-virtualmfadevicecreate (p. 125)
- iam-virtualmfadevicedel (p. 127)

iam-useruploadpolicy

Description

Takes a policy you've written and attaches it to the specified user. If a policy with that name is already attached to the user, it's overwritten with the new one. The command accepts either a string representing the policy, or a file containing the policy. For information about the contents of policies, refer to *Using AWS Identity and Access Management*.

A user can have only a limited number of policies. For more information, see Appendix A: Limitations on IAM Entities (p. 131).

Syntax

```
iam-useruploadpolicy -u USERNAME -p POLICYNAME [-f POLICYDOCUMENTFILE | -o POLICYDOCUMENT]
```

Options

Name	Description	Required
-u *USERNAME*	Name of the user the policy is for. Type: String Default: None	Yes
-p *POLICYNAME*	Name you want to assign the policy. Type: String Default: None	Yes
-f *POLICYDOCUMENTFILE*	Path and name of the file containing the policy. Type: String Condition: Either `-f POLICYDOCUMENTFILE` or `-o POLICYDOCUMENT` is required, but not both. Default: None	Conditional
-o *POLICYDOCUMENT*	The policy (a JSON text string). Type: String Condition: Either `-f POLICYDOCUMENTFILE` or `-o POLICYDOCUMENT` is required, but not both. Default: None	Conditional

Output

If the command is successful, the output is empty.

Example

The following example adds (or updates) the policy named KeyPolicy for the user named Bob. The policy is uploaded as a text string.

```
PROMPT> iam-useruploadpolicy -u Bob -p KeyPolicy -o {"Statement":[{"Effect":
"Allow","Action":["iam:*AccessKey*","iam:*SigningCertificate*"],"Resource":
"arn:aws:iam::123456789012:user/division_abc/subdivision_xyz/Bob"}]}
```

The following example adds (or updates) the policy named KeyPolicy for the user named Bob. The policy is uploaded as a text file.

```
PROMPT> iam-useruploadpolicy -u Bob -p KeyPolicy -f C:\Policies\KeyPo
licy_file.txt
```

Related Commands

- iam-useraddpolicy (p. 67)
- iam-userdelpolicy (p. 79)
- iam-userlistpolicies (p. 95)

iam-servercertdel

Description

Deletes the specified server certificate.

Important

If you are using a server certificate with Elastic Load Balancing, deleting the certificate could have implications for your application. If Elastic Load Balancing doesn't detect the deletion of bound certificates, it may continue to use the certificates. This could cause Elastic Load Balancing to stop accepting traffic. We recommend that you remove the reference to the certificate from Elastic Load Balancing before using this command to delete the certificate. For more information, go to DeleteLoadBalancerListeners in the *Elastic Load Balancing API Reference*.

Syntax

```
iam-servercertdel -s SERVERCERTNAME
```

Options

Name	Description	Required
`-s` *SERVERCERTNAME*	Name of the server certificate to delete. Type: String Default: None	Yes

Output

If the command is successful, the output is empty.

Examples

The following example deletes the server certificate with the name `ProductionCert`.

```
PROMPT> iam-servercertdel -s ProductionCert
```

Related Commands

- iam-servercertupload (p. 114)
- iam-servercertgetattributes (p. 109)
- iam-servercertmod (p. 112)
- iam-servercertlistbypath (p. 110)

iam-servercertgetattributes

Description

Returns the ARN and GUID of the server certificate. For more information about ARNs and GUIDs, go to Identifiers for IAM Entities in *Using AWS Identity and Access Management*.

Syntax

iam-servercertgetattributes -s *SERVERCERTNAME*

Options

Name	Description	Required
`-s` *SERVERCERTNAME*	Name of the server certificate you want to get information about. Type: String Default: None	Yes

Output

The output lists the ARN and GUID of the server certificate.

Examples

The following example returns output for a server certificate called ProdServerCert. The first line is the ARN and the second line is the GUID.

```
PROMPT> iam-servercertgetattributes -s ProdServerCert
arn:aws:iam::123456789012:server-certificate/company/servercerts/ProdServerCert
ASCACexample6TL7ZHQA
```

Related Commands

- iam-servercertupload (p. 114)
- iam-servercertdel (p. 108)
- iam-servercertmod (p. 112)
- iam-servercertlistbypath (p. 110)

iam-servercertlistbypath

Description

Lists the server certificates that have the specified path prefix. If there are none, the action returns an empty list. You can paginate the results using the `MaxItems` and `Marker` options.

Syntax

```
iam-servercertlistbypath [-p PATHPREFIX]
```

Options

Name	Description	Required
`-p` *`PATHPREFIX`*	The path prefix for filtering the results. For example, `/division_abc/subdivision_xyz/` would get all server certificates whose path starts with /division_abc/subdivision_xyz/. Type: String Default: / Condition: Provide this option only if you want to list the server certificates with a specific path prefix.	Conditional
`-i` *`MAXITEMS`*	Use this option only when paginating results to indicate the maximum number of items you want in the response. If there are additional items beyond the maximum you specify, the `IsTruncated` response element is `true`. Type: String Default: None	No
`-m` *`MARKER`*	Use this only when paginating results, and only in a subsequent request after you've received a response where the results are truncated. Set it to the value of the Marker element in the response you just received. Type: String Default: None	No

Output

The output lists the Amazon Resource Names (ARNs) for the server certificates that have the specified path prefix.

Examples

The following example lists all the server certificates in the specified path.

```
PROMPT> iam-servercertlistbypath -p /admin
arn:aws:iam::123456789012:server-certificate/admin/sns/bob
arn:aws:iam::123456789012:server-certificate/admin/elb/users/thomas
```

This example lists all server certificates.

```
PROMPT> iam-servercertlistbypath
arn:aws:iam::123456789012:server-certificate/noname@admin
arn:aws:iam::123456789012:server-certificate/users/bob
arn:aws:iam::123456789012:server-certificate/admin/sns/bob
arn:aws:iam::123456789012:server-certificate/admin/elb/users/thomas
arn:aws:iam::123456789012:server-certificate/customer/susan
```

Related Commands

- iam-servercertupload (p. 114)
- iam-servercertdel (p. 108)
- iam-servercertmod (p. 112)
- iam-servercertgetattributes (p. 109)

iam-servercertmod

Description

Changes the server certificate name or path (or both).

Important

You need to understand the implications of changing a server certificate path or name. For more information, see Managing Server Certificates in *Using AWS Identity and Access Management*.

Note

To change a server certificate name, the requester must have appropriate permissions on both the source object and the target object. For example, to change the name from ProductionCert to ProdCert, the entity making the request must have permission on ProductionCert and ProdCert, or must have permission on all (*). For more information about permissions, see Permissions and Policies.

Syntax

```
iam-servercertmod -s SERVERCERTNAME [-n NEWSERVERCERTNAME] [-p NEWPATH]
```

Options

Name	Description	Required
`-s` *SERVERCERTNAME*	Name of the server certificate to update. If you're changing the server certificate's name, this is the original name. Type: String Default: None	Yes
`-n` *NEWSERVERCERTIFICATENAME*	New name for the server certificate. Type: String Default: None Condition: Include this option only if you want to change the server certificate name.	Conditional
`-p` *NEWPATH*	New path for the server certificate. Type: String Default: None Condition: Include this option only if you want to change the server certificate path.	Conditional

Output

If the command is successful, the output is empty.

Examples

The following example changes the name of the server certificate from ProductionCert to ProdCert.

```
PROMPT> iam-servercertmod -s ProductionCert -n ProdCert
```

The following example changes the path for ProductionCert to `/division_abc/subdivision_xyz/test/`.

```
PROMPT> iam-servercertmod -s ProductionCert -p /division_abc/subdivi
sion_xyz/test/
```

Related Commands

- iam-servercertupload (p. 114)
- iam-servercertdel (p. 108)
- iam-servercertlistbypath (p. 110)
- iam-servercertgetattributes (p. 109)

iam-servercertupload

Description

Uploads a server certificate entity for the AWS account. The server certificate entity includes a public key certificate, a private key, and an optional certificate chain, which should all be PEM-encoded.

For information about the number of server certificates you can upload, see Appendix A: Limitations on IAM Entities (p. 131).

Syntax

```
iam-servercertupload -b CERTBODYFILE -k PRIVATEKEYFILE -s SERVERCERTNAME [-p
PATH] [-c CERTCHAINFILE] [-v]
```

Options

Name	Description	Required
`-b` *`CERTBODYFILE`*	The contents of the public key certificate in PEM-encoded format. Type: String Default: None	Yes
`-k` *`PRIVATEKEYFILE`*	The contents of the private key in PEM-encoded format. Type: String Default: None	Yes
`-s` *`SERVERCERTNAME`*	Name of the server certificate. Type: String Default: None	Yes
`-p` *`PATH`*	The path for the server certificate. For more information about paths, go to Identifiers for IAM Entities in *Using AWS Identity and Access Management*. This option is optional. If it is not included, the path defaults to a slash (/). Type: String Default: None	No
`-c` *`CERTCHAINFILE`*	The contents of the certificate chain. This is typically a concatenation of the chain's PEM-encoded public key certificates. Type: String Default: None	No

Output

If the command is successful, the output is empty. In verbose mode, IAM returns the server certificate Amazon Resource Name (ARN) and GUID.

Example

The following example uploads a server certificate and its associated private key, and names the server certificate ProductionCert in verbose mode.

```
PROMPT> iam-servercertupload -b C:\certs\ProductionCert.pem -k C:\keys\pri-
key.pem -s ProductionCert -v
arn:aws:iam::123456789012:server-certificate/ProductionCert
ASCACexampleKEZUQ4K
```

Related Commands

- iam-servercertmod (p. 112)
- iam-servercertdel (p. 108)
- iam-servercertlistbypath (p. 110)
- iam-servercertgetattributes (p. 109)

iam-accountaliascreate

Description

Creates an alias for your AWS account ID in the URL of your IAM-enabled AWS Management Console sign-in page. For information about using an AWS account alias, see Using an Alias for Your AWS Account ID in *Using AWS Identity and Access Management*.

Note

The URL containing the alias does not replace the original URL that contains your account ID. The original URL will remain active.

Note

You can create only a limited number of aliases, and use only a restricted set of characters for your alias. For more information, see Appendix A: Limitations on IAM Entities (p. 131).

Syntax

```
iam-accountaliascreate -a ACCOUNT ALIAS
```

Options

Name	Description	Required
`-a` *ACCOUNT ALIAS*	The alias for the AWS account ID. Type: String Default: None	Yes

Output

If the command is successful, the output is empty.

Example

The following example request creates the alias mycompany for the AWS account ID in URL of the AWS Management Console sign-in page.

```
PROMPT> iam-accountaliascreate -a mycompany
```

Related Commands

- iam-accountaliasdelete (p. 117)
- iam-accountaliaslist (p. 118)

iam-accountaliasdelete

Description

Deletes the alias in the URL for your IAM-enabled AWS Management Console sign-in page. For information about using an AWS account alias, see Using an Alias for Your AWS Account ID in *Using AWS Identity and Access Management*.

Syntax

```
iam-accountaliasdelete -a ACCOUNT ALIAS
```

Options

Name	Description	Required
-a *ACCOUNT ALIAS*	The alias you want to delete. Type: String Default: None	Yes

Output

If the command is successful, the output is empty.

Example

The following example request deletes the alias mycompany for the AWS account ID in the URL of the AWS Management Console sign-in page.

```
PROMPT> iam-accountaliasdelete -a mycompany
```

Related Commands

- iam-accountaliascreate (p. 116)
- iam-accountaliaslist (p. 118)

iam-accountaliaslist

Description

Lists aliases for the AWS account ID making the request. You can paginate the results using the `MaxItems` and `Marker` options. For information about using an AWS account alias, see Using an Alias for Your AWS Account ID in *Using AWS Identity and Access Management*.

Note

You can create only a limited number of aliases, and use only a restricted set of characters for your alias. For more information, see Appendix A: Limitations on IAM Entities (p. 131).

Syntax

```
iam-accountaliaslist
```

Options

Name	Description	Required
`-i` *`MAXITEMS`*	Use this option only when paginating results to indicate the maximum number of items you want in the response. If there are additional items beyond the maximum you specify, the `IsTruncated` response element is `true`. Type: String Default: None	No
`-m` *`MARKER`*	Use this only when paginating results, and only in a subsequent request after you've received a response where the results are truncated. Set it to the value of the Marker element in the response you just received. Type: String Default: None	No

Output

If the command is successful, it returns a list of aliases for the AWS account ID making the request, as well as the URL for the IAM-enabled sign-in page.

Example

The following example request lists the aliases for the AWS account ID making the request, and the URL for the AWS account sign-in page.

```
PROMPT> iam-accountaliaslist
Alias: MySite
Direct Signin Link: mysite.signin.aws.amazon.com
```

Related Commands

- iam-accountaliascreate (p. 116)
- iam-accountaliasdelete (p. 117)

iam-accountdelpasswordpolicy

Description

Removes the password policy for the account. For information about using password policies, go to Managing an IAM Password Policy in *Using AWS Identity and Access Management*.

Syntax

`iam-accountdelpasswordpolicy`

Output

If the command is successful, the output is empty.

Example

The following example request removes the password policy for the AWS account.

```
PROMPT> iam-accountdelpasswordpolicy
```

Related Commands

- iam-accountgetpasswordpolicy (p. 121)
- iam-accountmodpasswordpolicy (p. 123)

iam-accountgetpasswordpolicy

Description

Gets the password policy for the account. For information about using password policies, go to Managing an IAM Password Policy in *Using AWS Identity and Access Management*.

Syntax

`iam-accountgetpasswordpolicy`

Output

If the command is successful, the account password policy is returned.

Example

The following example request gets the password policy for the AWS account. The response shows that the minimum password length is 6 characters, non-alphanumeric characters and numbers are not required, and uppercase and lowercase characters are required. It also shows that allowing users to change their own password is enabled.

```
PROMPT> iam-accountgetpasswordpolicy
{MinimumPasswordLength: 6, RequireSymbols: false, RequireNumbers: false,
RequireUppercaseCharacters: true, RequireLowercaseCharacters: true,
AllowUsersToChangePassword: true, }
```

Related Commands

- iam-accountdelpasswordpolicy (p. 120)
- iam-accountmodpasswordpolicy (p. 123)

iam-accountgetsummary

Description

Retrieves account level information about entity usage and quotas. For more information about limits on IAM entities, see Appendix A: Limitations on IAM Entities (p. 131).

Syntax

`iam-accountgetsummary`

Output

If the command is successful, the command returns a list of entity quotas and the numbers of entities used.

Example

The following example request lists the IAM quotas and entities used for the AWS account ID making the request.

```
PROMPT> iam-accountgetsummary
Groups: 0
Users: 3
UsersQuota: 150
GroupsQuota: 50
GroupPolicySizeQuota: 10240
AccessKeysPerUserQuota: 2
UserPolicySizeQuota: 10240
GroupsPerUserQuota: 10
ServerCertificates: 0
SigningCertificatesPerUserQuota: 2
ServerCertificatesQuota: 10
AccountMFAEnabled: 0
MFADevicesInUse: 0
MFADevices: 20
```

iam-accountmodpasswordpolicy

Description

Updates the password policy for the account. For information about using password policies, go to Managing an IAM Password Policy in *Using AWS Identity and Access Management*.

Syntax

```
iam-accountmodpasswordpolicy [-l] [-m MINIMUMPASSWORDLENGTH] [-n] [-s] [-u]
[-a]
```

Options

Name	Description	Required
`-l` *REQUIRELOWERCASE*	The new password must contain at least one lowercase character. Type: String Default: None	No
`-m` *MINIMUMPASSWORDLENGTH*	The minimum number of required characters. Type: String Default: 6	No
`-n` *REQUIRENUMBERS*	The new password must contain at least one number. Type: String Default: None	No
`-s` *REQUIRESYMBOLS*	The new password must contain at least one symbol. Type: String Default: None	No
`-u` *REQUIREUPPERCASE*	The new password must contain at least one uppercase character. Type: String Default: None	No
`-a` *ALLOWUSERSTOCHANGEPASSWORD*	Allow all IAM users to change their password. Type: String Default: None	No

Output

If the command is successful, the output is empty.

Example

The following example request creates a password policy that requires new passwords to include a minimum of 8 characters, at least one uppercase letter, and at least one number. The example also enables all IAM users to change their own password.

```
PROMPT> iam-accountmodpasswordpolicy -m 8 -u -n -a
```

Related Commands

- iam-accountgetpasswordpolicy (p. 121)
- iam-accountdelpasswordpolicy (p. 120)

iam-virtualmfadevicecreate

Description

Generates seed information and a serial number in the form of an Amazon Resource Name (ARN) for a user's virtual MFA. For more information about virtual MFA devices, go to Using a Virtual MFA Device in *Using AWS Identity and Access Management*. For more information about ARNs, Identifiers for IAM Entities in *Using AWS Identity and Access Management*.

Syntax

```
iam-virtualmfadevicecreate -b BOOTSTRAPMETHOD -o OUTPUTFILE -p PATH -s
VIRTUALMFANAME
```

Options

Name	Description	Required
`-b BOOTSTRAPMETHOD`	Method to use to seed the virtual MFA. Type: String Valid Values: `QRCodePNG` | `Base32String` Default: None	Yes
`-o OUTPUTFILE`	The output path and file name. If you do not specify a value for `-o`, then the output will be to the command line if Base32String, and to a Java window image if QRCodePNG. Type: String Default: None	No
`-p PATH`	The path for the virtual MFA. If this value is not specified, it defaults to /. For more information about paths, go to Identifiers for IAM Entities in *Using AWS Identity and Access Management*. Type: String Default: None	No
`-s VIRTUALMFANAME`	The name for the virtual MFA. Type: String Default: None	Yes

Output

The output lists the serial number in the form of an ARN for the virtual MFA device.

Example

The following example creates a virtual MFA device with the name *yourMFAname* and using the QR code bootstrap method to seed the device. The QR code graphic file will be output to the file name and path *C:\company\mfa\yourMFAname.png*. For information about QR codes and seeding the virtual MFA device, go to Using a Virtual MFA Device in *Using AWS Identity and Access Management*.

```
PROMPT> iam-virtualmfadevicecreate -b QRCodePNG -s yourMFAname -o C:\com
pany\mfa\yourMFAname.png
```

Related Commands

- iam-userenablemfadevice (p. 80)
- iam-userresyncmfadevice (p. 104)
- iam-userlistmfadevices (p. 93)
- iam-userdeactivatemfadevice (p. 73)
- iam-virtualmfadevicedel (p. 127)

iam-virtualmfadevicedel

Description

Permanently deletes a virtual MFA for a user. For more information about virtual MFA devices, go to Using a Virtual MFA Device in *Using AWS Identity and Access Management*.

Note

You must deactivate a user's virtual MFA device before you can delete it. For information about deactivating MFA devices, see iam-userdeactivatemfadevice (p. 73).

Syntax

```
iam-virtualmfadevicedel -s SERIAL
```

Options

Name	Description	Required
`-s SERIAL`	The serial number for the virtual MFA device to delete. Type: String Default: None	Yes

Output

If the command is successful, the output is empty.

Example

The following example deletes the virtual MFA device with the serial number *arn:aws:iam::123456789012:mfa/yourMFAname*.

```
PROMPT> iam-virtualmfadevicedel arn:aws:iam::123456789012:mfa/yourMFAname
```

Related Commands

- iam-userenablemfadevice (p. 80)
- iam-userresyncmfadevice (p. 104)
- iam-userlistmfadevices (p. 93)
- iam-userdeactivatemfadevice (p. 73)
- iam-virtualmfadevicecreate (p. 125)

iam-virtualmfadevicelist

Description

Lists the virtual MFA devices under the AWS account by assignment status. If you do not specify an assignment status, the command returns a list of all virtual MFA devices. Assignment status can be `Assigned`, `Unassigned`, or `Any`. You can paginate the results using the `MaxItems` and `Marker` options.

For more information about virtual MFA devices, go to Using a Virtual MFA Device in *Using AWS Identity and Access Management*.

Syntax

```
iam-virtualmfadevicelist -t ASSIGNMENTSTATUS [-v]
```

Options

Name	Description	Required
`-t` *`ASSIGNMENTSTATUS`*	The type of virtual MFA device to list. Type: String Valid Values: `Any` | `Assigned` | `Unassigned`	Yes
`-v`	If the virtual MFA device is assigned, displays the device serial number, the ARN of the user the device is assigned to, and the user ID. If the device is assigned at the root account level, the user ID is the account number. If the device is unassigned, only the MFA device serial number is returned. Type: String Default: None	No
`-i` *`MAXITEMS`*	Use this option only when paginating results to indicate the maximum number of items you want in the response. If there are additional items beyond the maximum you specify, the `IsTruncated` response element is `true`. Type: String Default: None	No
`-m` *`MARKER`*	Use this only when paginating results, and only in a subsequent request after you've received a response where the results are truncated. Set it to the value of the Marker element in the response you just received. Type: String Default: None	No

Output

The output contains a list of virtual MFA devices associated with the AWS account.

Examples

The following example lists the virtual MFA devices associated with the account, where there are three devices associated with the account and `ASSIGNMENTSTATUS` is `Any`. The first serial number is for an unassigned device, the second serial number is for a device assigned at the root account level, the third serial number is for a device assigned to a user under the account named *ExampleUser*.

```
PROMPT> iam-virtualmfadevicelist -t Any –v
arn:aws:iam::123456789012:mfa/ExampleMFAdeviceName
arn:aws:iam::123456789012:mfa/RootMFAdeviceName
arn:aws:iam::123456789012:root
123456789012
arn:aws:iam::123456789012:mfa/ExampleUserMFAdeviceName
arn:aws:iam::123456789012:user/ExampleUser
AIDACKCEVSQ6C2EXAMPLE
IsTruncated: false
```

Related Commands

- iam-userenablemfadevice (p. 80)
- iam-userresyncmfadevice (p. 104)
- iam-userlistmfadevices (p. 93)
- iam-userdeactivatemfadevice (p. 73)
- iam-virtualmfadevicecreate (p. 125)
- iam-virtualmfadevicedel (p. 127)

Appendices

Topics

These appendices include additional information about IAM, such as limitations on entities and related resources.

For definitions of AWS terms, go to the Amazon Web Services Glossary.

Appendix A: Limitations on IAM Entities

This section lists restrictions on IAM entities, and describes how to get information about entity usage and quotas.

Note

To retrieve account level information about entity usage and quotas, use the GetAccountSummary API action or the iam-accountgetsummary CLI command.

Following are restrictions on names:

- Names of users, groups, roles, instance profiles, and server certificates must be alphanumeric, including the following common characters: plus (+), equal (=), comma (,), period (.), at (@), and dash (-).
- Path names must begin with a forward slash (/).
- Policy names must be unique to the user, group, or role they are attached to, and can contain any Basic Latin (ASCII) characters, minus the following reserved characters: backward slash (\), forward slash (/), asterisk (*), question mark (?), and white space. These characters are reserved according to RFC 3986 (for more information, see http://www.ietf.org/rfc/rfc3986.txt).
- User passwords (login profiles) can contain any Basic Latin (ASCII) characters.
- AWS account ID aliases must be unique across AWS products, and must be alphanumeric following DNS naming conventions. An alias must be lowercase, it must not start or end with a hyphen, it cannot contain two consecutive hyphens, and it cannot be a 12 digit number.

For a list of Basic Latin (ASCII) characters, go to the Library of Congress Basic Latin (ASCII) Code Table.

Names for entities are case sensitive and must be unique within the scope of your AWS account (regardless of the path you might give the entity).

Following are the default maximums for your entities:

- Groups per AWS account: 100
- Users per AWS account: 5000
  If you need to add a large number of users, consider using temporary security credentials. For more information about temporary security credentials, go to Using Temporary Security Credentials.
- Roles per AWS account: 250
- Instance Profiles per AWS account: 100
- Number of groups per user: 10 (that is, the user can be in this many groups)
- Access keys per user: 2
- Signing certificates per user: 2
- MFA devices in use per user: 1
- MFA devices in use per AWS account (at the root account level): 1
- Virtual MFA devices (assigned or unassigned) per AWS account: equal to the user quota for the account
- Server certificates per AWS account: 10
- AWS account aliases per AWS account: 1
- Login profiles per user: 1

You can request to increase these quotas for your AWS account on the IAM Limit Increase Contact Us Form.

Following are the maximum lengths for entities:

- Path: 512 characters
- User name: 64 characters
- Group name: 128 characters
- Role name: 64 characters
- Instance profile name: 128 characters
- GUID (applicable to users, groups, roles, and server certificates): 32 characters
- Policy name: 128 characters
- Certificate ID: 128 characters
- Login profile password: 1 to 128 characters
- AWS account ID alias: 3 to 63 characters.
- Total aggregate policy size per user or role: 2,048 characters (that is, you can have as many policies as you want for a given user or role as long as the sum size of the policies doesn't exceed 2,048 characters)
- Total aggregate policy size per group: 10,240 characters (that is, you can have as many policies as you want for a given group as long as the sum size of the policies doesn't exceed 10,240 characters)

Appendix B: IAM Resources

The following table lists related resources that you'll find useful as you work with this service.

Resource	Description
AWS Identity and Access Management Getting Started Guide	Provides instructions for using the service for the first time.
Using AWS Identity and Access Management	Describes how to use the service and all its features through the AWS Management Console, the CLI, or API.
AWS Identity and Access Management API Reference	Gives the WSDL and schema location; complete descriptions of the API actions, parameters, and data types; and a list of errors that the service returns.
AWS Identity and Access Management Quick Reference Card	Gives a concise listing of the commands you use with the CLI.
Product Information for IAM	The primary web page for information about IAM.
IAM Release Notes	The release notes give a high-level overview of the current release. They specifically note any new features, corrections, and known issues.
AWS Developer Resource Center	A central starting point to find documentation, code samples, release notes, and other information to help you build innovative applications with AWS.
IAM Discussion Forum	A community-based forum for developers to discuss technical questions related to IAM.
AWS Support Center	The home page for AWS Technical Support, including access to AWS developer forums, technical FAQs, service health dashboard, and premium support (if you are subscribed to this program).
AWS Premium Support Information	The primary web page for information about AWS Premium Support, a one-on-one, fast-response support channel to help you build and run applications on AWS Infrastructure Services.
Contact Us	A central contact point for inquiries concerning AWS billing, your AWS account, events, abuse, etc.
Conditions of Use	Detailed information about copyright and trademark usage at Amazon.com and other topics.