AWS Identity and Access Management

API Reference

API Version 2010-05-08

AWS Identity and Access Management: API Reference

Welcome

This guide provides descriptions of the Identity and Access Management (IAM) API as well as links to related content in the guide, Using IAM.

IAM is a web service that enables 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 (IAM). For information about setting up signatures and authorization through the API, go to Signing AWS API Requests in the *AWS General Reference*. For general information about using the Query API with IAM, go to Making Query Requests in *Using IAM*.

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

This document was last updated on July 17, 2012.

Actions

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

The following actions are supported:

AddRoleToInstanceProfile

Description

Adds the specified role to the specified instance profile. For more information about roles, go to Working with Roles. For more information about instance profiles, go to About Instance Profiles.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
InstanceProfileName	Name of the instance profile to update. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes
RoleName	Name of the role to add. Type: String Length constraints: Minimum length of 1. Maximum length of 64.	Yes

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
EntityAlreadyExists	The request was rejected because it attempted to create a resource that already exists.	409
LimitExceeded	The request was rejected because it attempted to create resources beyond the current AWS account limits. The error message describes the limit exceeded.	409
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=AddRoleToInstanceProfile
&InstanceProfileName=Webserver
&RoleName=S3Access
&Version=2010-05-08
```

```
&AUTHPARAMS
```

Sample Response

```
<AddRoleToInstanceProfileResponse xmlns="https://iam.amazonaws.com/doc/2010-05-
08/">
  <ResponseMetadata>
    <RequestId>12657608-99f2-11e1-a4c3-27EXAMPLE804</RequestId>
  </ResponseMetadata>
</AddRoleToInstanceProfileResponse>
```

AddUserToGroup

Description

Adds the specified user to the specified group.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
GroupName	Name of the group to update. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes
UserName	Name of the user to add. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
LimitExceeded	The request was rejected because it attempted to create resources beyond the current AWS account limits. The error message describes the limit exceeded.	409
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=AddUserToGroup
&GroupName=Managers
&UserName=Bob
&AUTHPARAMS
```

Sample Response

```
<AddUserToGroupResponse>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</AddUserToGroupResponse>
```

ChangePassword

Description

Changes the password of the IAM user calling `ChangePassword`. The root account password is not affected by this action. For information about modifying passwords, see Managing Passwords.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
NewPassword	Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes
OldPassword	Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
InvalidUserType	The request was rejected because the type of user for the transaction was incorrect.	400
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=ChangePassword
&OldPassword=U79}kgds4?
&NewPassword=Lb0*1(9xpN
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<ChangePasswordResponse>
```

```
    <ResponseMetadata>
        <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
    </ResponseMetadata>
</ChangePasswordResponse>
```

CreateAccessKey

Description

Creates a new AWS Secret Access Key and corresponding AWS Access Key ID for the specified user. The default status for new keys is `Active`.

If you do not specify a user name, IAM determines the user name implicitly based on the AWS Access Key ID signing the request. Because this action works for access keys under the AWS account, you can use this API to manage root credentials even if the AWS account has no associated users.

For information about limits on the number of keys you can create, see Limitations on IAM Entities in *Using AWS Identity and Access Management*.

> ⚠️ **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.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
UserName	The user name that the new key will belong to. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	No

Response Elements

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

Name	Description
AccessKey	Information about the access key. Type: AccessKey (p. 151)

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
LimitExceeded	The request was rejected because it attempted to create resources beyond the current AWS account limits. The error message describes the limit exceeded.	409

Error	Description	HTTP Status Code
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=CreateAccessKey
&UserName=Bob
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<CreateAccessKeyResponse>
   <CreateAccessKeyResult>
     <AccessKey>
         <UserName>Bob</UserName>
         <AccessKeyId>AKIAIOSFODNN7EXAMPLE</AccessKeyId>
         <Status>Active</Status>
         <SecretAccessKey>wJalrXUtnFEMI/K7MDENG/bPxRfiCYzEXAMPLEKEY
         </SecretAccessKey>
     </AccessKey>
   </CreateAccessKeyResult>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</CreateAccessKeyResponse>
```

CreateAccountAlias

Description

This action creates an alias for your AWS account. For information about using an AWS account alias, see Using an Alias for Your AWS Account ID in *Using AWS Identity and Access Management*.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
AccountAlias	Name of the account alias to create. Type: String Length constraints: Minimum length of 3. Maximum length of 63.	Yes

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
EntityAlreadyExists	The request was rejected because it attempted to create a resource that already exists.	409

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=CreateAccountAlias
&AccountAlias=foocorporation
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<CreateAccountAliasResponse>
  <ResponseMetadata>
    <RequestId>36b5db08-f1b0-11df-8fbe-45274EXAMPLE</RequestId>
  </ResponseMetadata>
</CreateAccountAliasResponse>
```

CreateGroup

Description

Creates a new group.

For information about the number of groups you can create, see Limitations on IAM Entities in *Using AWS Identity and Access Management*.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
GroupName	Name of the group to create. Do not include the path in this value. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes
Path	The path to the group. For more information about paths, see Identifiers for IAM Entities in *Using AWS Identity and Access Management*. This parameter is optional. If it is not included, it defaults to a slash (/). Type: String Length constraints: Minimum length of 1. Maximum length of 512.	No

Response Elements

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

Name	Description
Group	Information about the group. Type: Group (p. 160)

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
EntityAlreadyExists	The request was rejected because it attempted to create a resource that already exists.	409
LimitExceeded	The request was rejected because it attempted to create resources beyond the current AWS account limits. The error message describes the limit exceeded.	409
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=CreateGroup
&Path=/
&GroupName=Admins
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<CreateGroupResponse>
   <CreateGroupResult>
      <Group>
         <Path>/</Path>
         <GroupName>Admins</GroupName>
         <GroupId>AGPACKCEVSQ6C2EXAMPLE</GroupId>
         <Arn>arn:aws:iam::123456789012:group/Admins</Arn>
      </Group>
   </CreateGroupResult>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</CreateGroupResponse>
```

CreateInstanceProfile

Description

Creates a new instance profile. For information about instance profiles, go to About Instance Profiles.

For information about the number of instance profiles you can create, see Limitations on IAM Entities in *Using AWS Identity and Access Management*.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
InstanceProfileName	Name of the instance profile to create. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes
Path	The path to the instance profile. For more information about paths, see Identifiers for IAM Entities in *Using AWS Identity and Access Management*. This parameter is optional. If it is not included, it defaults to a slash (/). Type: String Length constraints: Minimum length of 1. Maximum length of 512.	No

Response Elements

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

Name	Description
InstanceProfile	Information about the instance profile. Type: InstanceProfile (p. 161)

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
EntityAlreadyExists	The request was rejected because it attempted to create a resource that already exists.	409
LimitExceeded	The request was rejected because it attempted to create resources beyond the current AWS account limits. The error message describes the limit exceeded.	409

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=CreateInstanceProfile
&InstanceProfileName=Webserver
&Path=/application_abc/component_xyz/
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<CreateInstanceProfileResponse xmlns="https://iam.amazonaws.com/doc/2010-05-
08/">
  <CreateInstanceProfileResult>
    <InstanceProfile>
      <InstanceProfileId>AIPAD5ARO2C5EXAMPLE3G</InstanceProfileId>
      <Roles/>
      <InstanceProfileName>Webserver</InstanceProfileName>
      <Path>/application_abc/component_xyz/</Path>
      <Arn>arn:aws:iam::123456789012:instance-profile/application_abc/compon
ent_xyz/Webserver</Arn>
      <CreateDate>2012-05-09T16:11:10.222Z</CreateDate>
    </InstanceProfile>
  </CreateInstanceProfileResult>
  <ResponseMetadata>
    <RequestId>974142ee-99f1-11e1-a4c3-27EXAMPLE804</RequestId>
  </ResponseMetadata>
</CreateInstanceProfileResponse>
```

CreateLoginProfile

Description

Creates a password for the specified user, giving the user the ability to access AWS services through the AWS Management Console. For more information about managing passwords, see Managing Passwords in *Using IAM*.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
Password	The new password for the user name. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes
UserName	Name of the user to create a password for. Type: String Length constraints: Minimum length of 1. Maximum length of 64.	Yes

Response Elements

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

Name	Description
`LoginProfile`	The user name and password create date. Type: LoginProfile (p. 169)

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
`EntityAlreadyExists`	The request was rejected because it attempted to create a resource that already exists.	409
`NoSuchEntity`	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404
`PasswordPolicyViolation`	The request was rejected because the provided password did not meet the requirements imposed by the account password policy.	400

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=CreateLoginProfile
&UserName=Bob
&Password=Password1
&AUTHPARAMS
```

Sample Response

```
<CreateLoginProfileResponse>
   <CreateUserResult>
      <LoginProfile>
         <UserName>Bob</UserName>
         <CreateDate>2011-09-19T23:00:56Z</CreateDate>
      </LoginProfile>
   </CreateUserResult>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</CreateLoginProfileResponse>
```

CreateRole

Description

Creates a new role for your AWS account. For more information about roles, go to Working with Roles. For information about limitations on role names and the number of roles you can create, go to Limitations on IAM Entities in *Using AWS Identity and Access Management*.

The policy grants permission to an EC2 instance to assume the role. The policy is URL-encoded according to RFC 3986. For more information about RFC 3986, go to http://www.faqs.org/rfcs/rfc3986.html. Currently, only EC2 instances can assume roles.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
`AssumeRolePolicyDocument`	The policy that grants an entity permission to assume the role. Type: String Length constraints: Minimum length of 1. Maximum length of 131072.	Yes
`Path`	The path to the role. For more information about paths, see Identifiers for IAM Entities in *Using AWS Identity and Access Management*. This parameter is optional. If it is not included, it defaults to a slash (/). Type: String Length constraints: Minimum length of 1. Maximum length of 512.	No
`RoleName`	Name of the role to create. Type: String Length constraints: Minimum length of 1. Maximum length of 64.	Yes

Response Elements

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

Name	Description
`Role`	Information about the role. Type: Role (p. 171)

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
`EntityAlreadyExists`	The request was rejected because it attempted to create a resource that already exists.	409
`LimitExceeded`	The request was rejected because it attempted to create resources beyond the current AWS account limits. The error message describes the limit exceeded.	409
`MalformedPolicyDocument`	The request was rejected because the policy document was malformed. The error message describes the specific error.	400

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=CreateRole
&RoleName=S3Access
&Path=/application_abc/component_xyz/
&AssumeRolePolicyDocument={"Version":"2008-10-17","Statement":[{"Effect":"Al
low","Principal":{"Service":["ec2.amazonaws.com"]},"Action":["sts:AssumeRole"]}]}
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<CreateRoleResponse xmlns="https://iam.amazonaws.com/doc/2010-05-08/">
  <CreateRoleResult>
    <Role>
      <Path>/application_abc/component_xyz/</Path>
      <Arn>arn:aws:iam::123456789012:role/application_abc/component_xyz/S3Ac
cess</Arn>
      <RoleName>S3Access</RoleName>
      <AssumeRolePolicyDocument>{"Version":"2008-10-17","Statement":[{"Ef
fect":"Allow","Principal":{"Service":["ec2.amazonaws.com"]},"Action":["sts:As
sumeRole"]}]}</AssumeRolePolicyDocument>
      <CreateDate>2012-05-08T23:34:01.495Z</CreateDate>
      <RoleId>AROADBQP57FF2AEXAMPLE</RoleId>
    </Role>
  </CreateRoleResult>
  <ResponseMetadata>
    <RequestId>4a93ceee-9966-11e1-b624-b1aEXAMPLE7c</RequestId>
  </ResponseMetadata>
</CreateRoleResponse>
```

CreateUser

Description

Creates a new user for your AWS account.

For information about limitations on the number of users you can create, see Limitations on IAM Entities in *Using AWS Identity and Access Management*.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
Path	The path for the user name. For more information about paths, see Identifiers for IAM Entities in *Using AWS Identity and Access Management*. This parameter is optional. If it is not included, it defaults to a slash (/). Type: String Length constraints: Minimum length of 1. Maximum length of 512.	No
UserName	Name of the user to create. Type: String Length constraints: Minimum length of 1. Maximum length of 64.	Yes

Response Elements

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

Name	Description
User	Information about the user. Type: User (p. 175)

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
EntityAlreadyExists	The request was rejected because it attempted to create a resource that already exists.	409
LimitExceeded	The request was rejected because it attempted to create resources beyond the current AWS account limits. The error message describes the limit exceeded.	409
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=CreateUser
&Path=/division_abc/subdivision_xyz/
&UserName=Bob
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<CreateUserResponse>
   <CreateUserResult>
      <User>
         <Path>/division_abc/subdivision_xyz/</Path>
         <UserName>Bob</UserName>
         <UserId>AIDACKCEVSQ6C2EXAMPLE</UserId>
         <Arn>arn:aws:iam::123456789012:user/division_abc/subdivision_xyz/Bob
         </Arn>
      </User>
   </CreateUserResult>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</CreateUserResponse>
```

CreateVirtualMFADevice

Description

Creates a new virtual MFA device for the AWS account. After creating the virtual MFA, use EnableMFADevice to attach the MFA device to an IAM user. For more information about creating and working with virtual MFA devices, go to Using a Virtual MFA Device in *Using AWS Identity and Access Management*.

For information about limits on the number of MFA devices you can create, see Limitations on Entities in *Using AWS Identity and Access Management*.

⚠️ **Important**

The seed information contained in the QR code and the Base32 string should be treated like any other secret access information, such as your AWS access keys or your passwords. After you provision your virtual device, you should ensure that the information is destroyed following secure procedures.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
Path	The path for the virtual MFA device. For more information about paths, see Identifiers for IAM Entities in *Using AWS Identity and Access Management*. This parameter is optional. If it is not included, it defaults to a slash (/). Type: String Length constraints: Minimum length of 1. Maximum length of 512.	No
VirtualMFADeviceName	The name of the virtual MFA device. Use with path to uniquely identify a virtual MFA device. Type: String Length constraints: Minimum length of 1.	Yes

Response Elements

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

Name	Description
VirtualMFADevice	A newly created virtual MFA device. Type: VirtualMFADevice (p. 176)

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
EntityAlreadyExists	The request was rejected because it attempted to create a resource that already exists.	409
LimitExceeded	The request was rejected because it attempted to create resources beyond the current AWS account limits. The error message describes the limit exceeded.	409

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=CreateVirtualMFADevice
&VirtualMFADeviceName=ExampleName
&Path=/
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<CreateVirtualMFADeviceResponse>
  <CreateVirtualMFADeviceResult>
    <VirtualMFADevice>
      <SerialNumber>arn:aws:iam::123456789012:mfa/ExampleName</SerialNumber>
      <Base32StringSeed>2K5K5XTLA7GGE75TQLYEXAMPLEEXAMPLEEXAMPLECHDFW4KJYZ6
      UFQ75LL7COCYKM</Base32StringSeed>
      <QRCodePNG>89504E470D0A1A0AASDFAHSDFKJKLJFKALSDFJASDF</QRCodePNG> <!--
byte array of png file -->
    </VirtualMFADevice>
  </CreateVirtualMFADeviceResult>
  <ResponseMetadata>
    <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
  </ResponseMetadata>
</CreateVirtualMFADeviceResponse>
```

DeactivateMFADevice

Description

Deactivates the specified MFA device and removes it from association with the user name for which it was originally enabled.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
SerialNumber	The serial number that uniquely identifies the MFA device. For virtual MFA devices, the serial number is the device ARN. Type: String Length constraints: Minimum length of 9. Maximum length of 256.	Yes
UserName	Name of the user whose MFA device you want to deactivate. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
EntityTemporarilyUnmodifiable	The request was rejected because it referenced an entity that is temporarily unmodifiable, such as a user name that was deleted and then recreated. The error indicates that the request is likely to succeed if you try again after waiting several minutes. The error message describes the entity.	409
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=DeactivateMFADevice
&UserName=Bob
&SerialNumber=R1234
```

```
&AUTHPARAMS
```

Sample Response

```
<DeactivateMFADeviceResponse>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</DeactivateMFADeviceResponse>
```

DeleteAccessKey

Description

Deletes the access key associated with the specified user.

If you do not specify a user name, IAM determines the user name implicitly based on the AWS Access Key ID signing the request. Because this action works for access keys under the AWS account, you can use this API to manage root credentials even if the AWS account has no associated users.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
AccessKeyId	The Access Key ID for the Access Key ID and Secret Access Key you want to delete. Type: String Length constraints: Minimum length of 16. Maximum length of 32.	Yes
UserName	Name of the user whose key you want to delete. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	No

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=DeleteAccessKey
&UserName=Bob
&AccessKeyId=AKIAIOSFODNN7EXAMPLE
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<DeleteAccessKeyResponse>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</DeleteAccessKeyResponse>
```

DeleteAccountAlias

Description

Deletes the specified AWS account alias. For information about using an AWS account alias, see Using an Alias for Your AWS Account ID in *Using AWS Identity and Access Management*.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
AccountAlias	Name of the account alias to delete. Type: String Length constraints: Minimum length of 3. Maximum length of 63.	Yes

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=DeleteAccountAlias
&AccountAlias=foocorporation
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<DeleteAccountAliasResponse>
  <ResponseMetadata>
    <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
  </ResponseMetadata>
</DeleteAccountAliasResponse>
```

DeleteAccountPasswordPolicy

Description

Deletes the password policy for the AWS account.

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=DeleteAccountPasswordPolicy
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<DeleteAccountPasswordPolicyResponse>
   <ResponseMetadata>
     <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
 </ResponseMetadata>
</DeleteAccountPasswordPolicy>
```

DeleteGroup

Description

Deletes the specified group. The group must not contain any users or have any attached policies.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
GroupName	Name of the group to delete. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
DeleteConflict	The request was rejected because it attempted to delete a resource that has attached subordinate entities. The error message describes these entities.	409
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=DeleteGroup
&Group=Test
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<DeleteGroupResponse>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
```

```
</DeleteGroupResponse>
```

DeleteGroupPolicy

Description

Deletes the specified policy that is associated with the specified group.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
GroupName	Name of the group the policy is associated with. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes
PolicyName	Name of the policy document to delete. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=DeleteGroupPolicy
&GroupName=Admins
&PolicyName=AdminRoot
&AUTHPARAMS
```

Sample Response

```
<DeleteGroupPolicyResponse>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
```

```
</DeleteGroupPolicyResponse>
```

DeleteInstanceProfile

Description

Deletes the specified instance profile. The instance profile must not have an associated role.

⚠️ **Important**

Make sure you do not have any Amazon EC2 instances running with the 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.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
InstanceProfileName	Name of the instance profile to delete. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
DeleteConflict	The request was rejected because it attempted to delete a resource that has attached subordinate entities. The error message describes these entities.	409
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=DeleteInstanceProfile
&InstanceProfileName=Webserver
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<DeleteInstanceProfileResponse xmlns="https://iam.amazonaws.com/doc/2010-05-
08/">
  <ResponseMetadata>
    <RequestId>90c18667-99f3-11e1-a4c3-27EXAMPLE804</RequestId>
  </ResponseMetadata>
</DeleteInstanceProfileResponse>
```

DeleteLoginProfile

Description

Deletes the password for the specified user, which terminates the user's ability to access AWS services through the AWS Management Console.

> ⚠️ **Important**
>
> Deleting a user's password does not prevent a user from accessing IAM through the command line interface or the API. To prevent all user access you must also either make the access key inactive or delete it. For more information about making keys inactive or deleting them, see UpdateAccessKey (p. 128) and DeleteAccessKey (p. 28).

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
UserName	Name of the user whose password you want to delete. Type: String Length constraints: Minimum length of 1. Maximum length of 64.	Yes

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
EntityTemporarilyUnmodifiable	The request was rejected because it referenced an entity that is temporarily unmodifiable, such as a user name that was deleted and then recreated. The error indicates that the request is likely to succeed if you try again after waiting several minutes. The error message describes the entity.	409
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=DeleteLoginProfile
&UserName=Bob
```

```
&AUTHPARAMS
```

Sample Response

```
<DeleteLoginProfileResponse>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</DeleteLoginProfileResponse>
```

DeleteRole

Description

Deletes the specified role. The role must not have any policies attached. For more information about roles, go to Working with Roles.

> ⚠️ **Important**
>
> Make sure you do not have any Amazon EC2 instances running with the role 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.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
RoleName	Name of the role to delete. Type: String Length constraints: Minimum length of 1. Maximum length of 64.	Yes

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
DeleteConflict	The request was rejected because it attempted to delete a resource that has attached subordinate entities. The error message describes these entities.	409
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=DeleteRole
&RoleName=S3Access
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<DeleteRoleResponse xmlns="https://iam.amazonaws.com/doc/2010-05-08/">
  <ResponseMetadata>
    <RequestId>913e3f37-99ed-11e1-a4c3-270EXAMPLE04</RequestId>
  </ResponseMetadata>
</DeleteRoleResponse>
```

DeleteRolePolicy

Description

Deletes the specified policy associated with the specified role.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
PolicyName	Name of the policy document to delete. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes
RoleName	Name of the role the associated with the policy. Type: String Length constraints: Minimum length of 1. Maximum length of 64.	Yes

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=DeleteRolePolicy
&PolicyName=S3AccessPolicy
&RoleName=S3Access
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<DeleteRolePolicyResponse xmlns="https://iam.amazonaws.com/doc/2010-05-08/">
  <ResponseMetadata>
    <RequestId>c749ee7f-99ef-11e1-a4c3-27EXAMPLE804</RequestId>
```

```
   </ResponseMetadata>
</DeleteRolePolicyResponse>
```

DeleteServerCertificate

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*.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
ServerCertificateName	The name of the server certificate you want to delete. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
DeleteConflict	The request was rejected because it attempted to delete a resource that has attached subordinate entities. The error message describes these entities.	409
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=DeleteServerCertificate
&ServerCertificateName=ProdServerCert
&Version=2010-05-08
```

```
&AUTHPARAMS
```

Sample Response

```
<DeleteServerCertificateResponse>
  <ResponseMetadata>
    <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
  </ResponseMetadata>
</DeleteServerCertificateResponse>
```

DeleteSigningCertificate

Description

Deletes the specified signing certificate associated with the specified user.

If you do not specify a user name, IAM determines the user name implicitly based on the AWS Access Key ID signing the request. Because this action works for access keys under the AWS account, you can use this API to manage root credentials even if the AWS account has no associated users.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
CertificateId	ID of the signing certificate to delete. Type: String Length constraints: Minimum length of 24. Maximum length of 128.	Yes
UserName	Name of the user the signing certificate belongs to. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	No

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

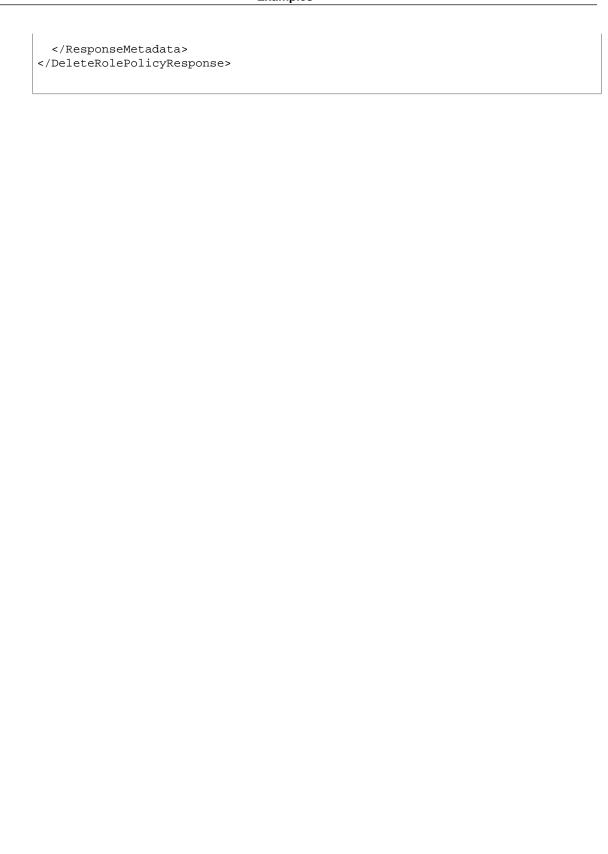| Error	Description	HTTP Status Code
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=DeleteSigningCertificate
&UserName=Bob
&CertificateId=TA7SMP42TDN5Z26OBPJE7EXAMPLE
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<DeleteSigningCertificateResponse>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</DeleteSigningCertificateResponse>
```

DeleteUser

Description

Deletes the specified user. The user must not belong to any groups, have any keys or signing certificates, or have any attached policies.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
UserName	Name of the user to delete. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
DeleteConflict	The request was rejected because it attempted to delete a resource that has attached subordinate entities. The error message describes these entities.	409
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=DeleteUser
&UserName=Bob
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<DeleteUserResponse>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
```

```
    </ResponseMetadata>
</DeleteUserResponse>
```

DeleteUserPolicy

Description

Deletes the specified policy associated with the specified user.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
PolicyName	Name of the policy document to delete. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes
UserName	Name of the user the policy is associated with. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=DeleteUserPolicy
&UserName=Bob
&PolicyName=AllAccessPolicy
&AUTHPARAMS
```

Sample Response

```
<DeleteUserPolicyResponse>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
```

```
</DeleteUserPolicyResponse>
```

DeleteVirtualMFADevice

Description

Deletes a virtual MFA device.

> 📓 **Note**
>
> You must deactivate a user's virtual MFA device before you can delete it. For information about deactivating MFA devices, see DeactivateMFADevice.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
SerialNumber	The serial number that uniquely identifies the MFA device. For virtual MFA devices, the serial number is the same as the ARN. Type: String Length constraints: Minimum length of 9. Maximum length of 256.	Yes

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
DeleteConflict	The request was rejected because it attempted to delete a resource that has attached subordinate entities. The error message describes these entities.	409
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=DeleteVirtualMFADevice
&SerialNumber=arn:aws:iam::123456789012:mfa/ExampleName
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<DeleteVirtualMFADeviceResponse>
  <DeleteVirtualMFADeviceResult>
    <VirtualMFADevice>
      <SerialNumber>arn:aws:iam::123456789012:mfa/ExampleName</SerialNumber>
    </VirtualMFADevice>
  </DeleteVirtualMFADeviceResult>
  <ResponseMetadata>
    <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
  </ResponseMetadata>
</DeleteVirtualMFADeviceResponse>
```

EnableMFADevice

Description

Enables the specified MFA device and associates it with the specified user name. When enabled, the MFA device is required for every subsequent login by the user name associated with the device.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
AuthenticationCode1	An authentication code emitted by the device. Type: String Length constraints: Minimum length of 6. Maximum length of 6.	Yes
AuthenticationCode2	A subsequent authentication code emitted by the device. Type: String Length constraints: Minimum length of 6. Maximum length of 6.	Yes
SerialNumber	The serial number that uniquely identifies the MFA device. For virtual MFA devices, the serial number is the device ARN. Type: String Length constraints: Minimum length of 9. Maximum length of 256.	Yes
UserName	Name of the user for whom you want to enable the MFA device. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
EntityAlreadyExists	The request was rejected because it attempted to create a resource that already exists.	409
EntityTemporarilyUnmodifiable	The request was rejected because it referenced an entity that is temporarily unmodifiable, such as a user name that was deleted and then recreated. The error indicates that the request is likely to succeed if you try again after waiting several minutes. The error message describes the entity.	409

Error	Description	HTTP Status Code
InvalidAuthenticationCode	The request was rejected because the authentication code was not recognized. The error message describes the specific error.	403
LimitExceeded	The request was rejected because it attempted to create resources beyond the current AWS account limits. The error message describes the limit exceeded.	409
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=EnableMFADevice
&UserName=Bob
&SerialNumber=R1234
&AuthenticationCode1=234567
&AuthenticationCode2=987654
&AUTHPARAMS
```

Sample Response

```
<EnableMFADeviceResponse>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</EnableMFADeviceResponse>
```

GetAccountPasswordPolicy

Description

Retrieves the password policy for the AWS account. For more information about using a password policy, go to Managing an IAM Password Policy.

Response Elements

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

Name	Description
PasswordPolicy	The PasswordPolicy data type contains information about the account password policy. This data type is used as a response element in the action GetAccountPasswordPolicy (p. 56). Type: PasswordPolicy (p. 170)

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=GetAccountPasswordPolicy
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<GetAccountPasswordPolicyResponse>
  <GetAccountPasswordPolicyResult>
    <PasswordPolicy>
        <MinimumPasswordLength>6</MinimumPasswordLength>,
        <RequireUppercaseCharacters>false</RequireUppercaseCharacters>
```

```
            <RequireLowercaseCharacters>false</RequireLowercaseCharacters>
            <RequireNumbers>false</RequireNumbers>
            <RequireSymbols>false</RequireSymbols>
            <AllowUsersToChangePassword>true</AllowUsersToChangePassword>
      </PasswordPolicy>
   </GetAccountPasswordPolicyResult>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</GetAccountPasswordPolicyResponse>
```

GetAccountSummary

Description

Retrieves account level information about account entity usage and IAM quotas.

For information about limitations on IAM entities, see Limitations on IAM Entities in *Using AWS Identity and Access Management*.

Response Elements

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

Name	Description
`SummaryMap`	A set of key value pairs containing account-level information. `SummaryMap` contains the following keys: • `AccessKeysPerUserQuota` - Maximum number of access keys that can be created per user • `AccountMFAEnabled` - 1 if the root account has an MFA device assigned to it, 0 otherwise • `AssumeRolePolicySizeQuota` - Maximum allowed size for assume role policy documents (in kilobytes) • `GroupPolicySizeQuota` - Maximum allowed size for Group policy documents (in kilobytes) • `Groups` - Number of Groups for the AWS account • `GroupsPerUserQuota` - Maximum number of groups a user can belong to • `GroupsQuota` - Maximum groups allowed for the AWS account • `InstanceProfiles` - Number of instance profiles for the AWS account • `InstanceProfilesQuota` - Maximum instance profiles allowed for the AWS account • `MFADevices` - Number of MFA devices, either assigned or unassigned • `MFADevicesInUse` - Number of MFA devices that have been assigned to an IAM user or to the root account • `RolePolicySizeQuota` - Maximum allowed size for role policy documents (in kilobytes) • `Roles` - Number of roles for the AWS account • `RolesQuota` - Maximum roles allowed for the AWS account • `ServerCertificates` - Number of server certificates for the AWS account • `ServerCertificatesQuota` - Maximum server certificates allowed for the AWS account • `SigningCertificatesPerUserQuota` - Maximum number of X509 certificates allowed for a user • `UserPolicySizeQuota` - Maximum allowed size for user policy documents (in kilobytes) • `Users` - Number of users for the AWS account • `UsersQuota` - Maximum users allowed for the AWS account Type: String to Integer map

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=GetAccountSummary
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<GetAccountSummaryResponse>
  <GetAccountSummaryResult>
    <SummaryMap>
      <entry>
        <key>Groups</key>
        <value>31</value>
      </entry>
      <entry>
        <key>GroupsQuota</key>
        <value>50</value>
      </entry>
      <entry>
        <key>UsersQuota</key>
        <value>150</value>
      </entry>
      <entry>
        <key>Users</key>
        <value>35</value>
      </entry>
      <entry>
        <key>GroupPolicySizeQuota</key>
        <value>10240</value>
      </entry>
      <entry>
        <key>AccessKeysPerUserQuota</key>
        <value>2</value>
      </entry>
      <entry>
        <key>GroupsPerUserQuota</key>
        <value>10</value>
      </entry>
      <entry>
        <key>UserPolicySizeQuota</key>
        <value>10240</value>
      </entry>
      <entry>
        <key>SigningCertificatesPerUserQuota</key>
        <value>2</value>
      </entry>
      <entry>
        <key>ServerCertificates</key>
```

```
          <value>0</value>
        </entry>
        <entry>
          <key>ServerCertificatesQuota</key>
          <value>10</value>
        </entry>
        <entry>
          <key>AccountMFAEnabled</key>
          <value>0</value>
        </entry>
        <entry>
          <key>MFADevicesInUse</key>
          <value>10</value>
        </entry>
        <entry>
          <key>MFADevices</key>
          <value>20</value>
        </entry>
     </SummaryMap>
  </GetAccountSummaryResult>
  <ResponseMetadata>
     <RequestId>f1e38443-f1ad-11df-b1ef-a9265EXAMPLE</RequestId>
  </ResponseMetadata>
</GetAccountSummaryResponse>
```

GetGroup

Description

Returns a list of users that are in the specified group. You can paginate the results using the `MaxItems` and `Marker` parameters.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
GroupName	Name of the group. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes
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 Length constraints: Minimum length of 1. Maximum length of 320.	No
MaxItems	Use this only when paginating results to indicate the maximum number of user names you want in the response. If there are additional user names beyond the maximum you specify, the `IsTruncated` response element is `true`. This parameter is optional. If you do not include it, it defaults to 100. Type: Integer	No

Response Elements

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

Name	Description
`Group`	Information about the group. Type: Group (p. 160)
`IsTruncated`	A flag that indicates whether there are more user names to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more user names in the list. Type: Boolean
`Marker`	If IsTruncated is `true`, then this element is present and contains the value to use for the `Marker` parameter in a subsequent pagination request. Type: String
`Users`	A list of users in the group. Type: User (p. 175) list

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=GetGroup
&GroupName=Admins
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<GetGroupResponse>
   <GetGroupResult>
      <Group>
         <Path>/</Path>
         <GroupName>Admins</GroupName>
         <GroupId>AGPACKCEVSQ6C2EXAMPLE</GroupId>
         <Arn>arn:aws:iam::123456789012:group/Admins</Arn>
      </Group>
      <Users>
         <member>
            <Path>/division_abc/subdivision_xyz/</Path>
            <UserName>Bob</UserName>
            <UserId>AIDACKCEVSQ6C2EXAMPLE</UserId>
            <Arn>
            arn:aws:iam::123456789012:user/division_abc/subdivision_xyz/Bob
            </Arn>
         </member>
         <member>
            <Path>/division_abc/subdivision_xyz/</Path>
            <UserName>Susan</UserName>
            <UserId>AIDACKCEVSQ6C2EXAMPLE</UserId>
            <Arn>
            arn:aws:iam::123456789012:user/division_abc/subdivision_xyz/Susan
            </Arn>
         </member>
      </Users>
      <IsTruncated>false</IsTruncated>
   </GetGroupResult>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
```

```
</GetGroupResponse>
```

GetGroupPolicy

Description

Retrieves the specified policy document for the specified group. The returned policy is URL-encoded according to RFC 3986. For more information about RFC 3986, go to http://www.faqs.org/rfcs/rfc3986.html.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
GroupName	Name of the group the policy is associated with. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes
PolicyName	Name of the policy document to get. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes

Response Elements

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

Name	Description
GroupName	The group the policy is associated with. Type: String
PolicyDocument	The policy document. Type: String
PolicyName	The name of the policy. Type: String

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=GetGroupPolicy
&GroupName=Admins
&PolicyName=AdminRoot
&AUTHPARAMS
```

Sample Response

```
<GetGroupPolicyResponse>
   <GetGroupPolicyResult>
      <GroupName>Admins</GroupName>
      <PolicyName>AdminRoot</PolicyName>
      <PolicyDocument>
      {"Statement":[{"Effect":"Allow","Action":"*","Resource":"*"}]}
      </PolicyDocument>
   </GetGroupPolicyResult>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</GetGroupPolicyResponse>
```

GetInstanceProfile

Description

Retrieves information about the specified instance profile, including the instance profile's path, GUID, ARN, and role. For more information about instance profiles, go to About Instance Profiles. For more information about ARNs, go to ARNs.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
InstanceProfileName	Name of the instance profile to get information about. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes

Response Elements

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

Name	Description
InstanceProfile	Information about the instance profile. Type: InstanceProfile (p. 161)

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=GetInstanceProfile
&InstanceProfileName=Webserver
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<GetInstanceProfileResponse xmlns="https://iam.amazonaws.com/doc/2010-05-08/">

  <GetInstanceProfileResult>
    <InstanceProfile>
      <InstanceProfileId>AIPAD5ARO2C5EXAMPLE3G</InstanceProfileId>
      <Roles>
        <member>
          <Path>/application_abc/component_xyz/</Path>
          <Arn>arn:aws:iam::123456789012:role/application_abc/component_xyz/S3Ac
cess</Arn>
          <RoleName>S3Access</RoleName>
          <AssumeRolePolicyDocument>{"Version":"2008-10-17","Statement":[{"Ef
fect":"Allow","Principal":{"Service":["ec2.amazonaws.com"]},"Action":["sts:As
sumeRole"]}]}</AssumeRolePolicyDocument>
          <CreateDate>2012-05-09T15:45:35Z</CreateDate>
          <RoleId>AROACVYKSVTSZFEXAMPLE</RoleId>
        </member>
      </Roles>
      <InstanceProfileName>Webserver</InstanceProfileName>
      <Path>/application_abc/component_xyz/</Path>
      <Arn>arn:aws:iam::123456789012:instance-profile/application_abc/compon
ent_xyz/Webserver</Arn>
      <CreateDate>2012-05-09T16:11:10Z</CreateDate>
    </InstanceProfile>
  </GetInstanceProfileResult>
  <ResponseMetadata>
    <RequestId>37289fda-99f2-11e1-a4c3-27EXAMPLE804</RequestId>
  </ResponseMetadata>
</GetInstanceProfileResponse>
```

GetLoginProfile

Description

Retrieves the user name and password create date for the specified user.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
UserName	Name of the user whose login profile you want to retrieve. Type: String Length constraints: Minimum length of 1. Maximum length of 64.	Yes

Response Elements

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

Name	Description
`LoginProfile`	User name and password create date for the user. Type: LoginProfile (p. 169)

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
`NoSuchEntity`	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=GetLoginProfile
&UserName=Bob
&AUTHPARAMS
```

Sample Response

```
<GetLoginProfileResponse>
   <GetLoginProfileResult>
      <LoginProfile>
         <UserName>Bob</UserName>
         <CreateDate>2011-09-19T23:00:56Z</CreateDate>
      </LoginProfile>
   </GetLoginProfileResult>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</GetLoginProfileResponse>
```

GetRole

Description

Retrieves information about the specified role, including the role's path, GUID, ARN, and the policy granting permission to EC2 to assume the role. For more information about ARNs, go to ARNs. For more information about roles, go to Working with Roles.

The returned policy is URL-encoded according to RFC 3986. For more information about RFC 3986, go to http://www.faqs.org/rfcs/rfc3986.html.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
RoleName	Name of the role to get information about. Type: String Length constraints: Minimum length of 1. Maximum length of 64.	Yes

Response Elements

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

Name	Description
`Role`	Information about the role. Type: Role (p. 171)

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
`NoSuchEntity`	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=GetRole
&RoleName=S3Access
&Version=2010-05-08
```

```
&AUTHPARAMS
```

Sample Response

```
<GetRoleResponse xmlns="https://iam.amazonaws.com/doc/2010-05-08/">
  <GetRoleResult>
    <Role>
      <Path>/application_abc/component_xyz/</Path>
      <Arn>arn:aws:iam::123456789012:role/application_abc/component_xyz/S3Ac
cess</Arn>
      <RoleName>S3Access</RoleName>
      <AssumeRolePolicyDocument>{"Version":"2008-10-17","Statement":[{"Ef
fect":"Allow","Principal":{"Service":["ec2.amazonaws.com"]},"Action":["sts:As
sumeRole"]}]}</AssumeRolePolicyDocument>
      <CreateDate>2012-05-08T23:34:01Z</CreateDate>
      <RoleId>AROADBQP57FF2AEXAMPLE</RoleId>
    </Role>
  </GetRoleResult>
  <ResponseMetadata>
    <RequestId>df37e965-9967-11e1-a4c3-270EXAMPLE04</RequestId>
  </ResponseMetadata>
</GetRoleResponse>
```

GetRolePolicy

Description

Retrieves the specified policy document for the specified role. For more information about roles, go to Working with Roles.

The returned policy is URL-encoded according to RFC 3986. For more information about RFC 3986, go to http://www.faqs.org/rfcs/rfc3986.html.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
PolicyName	Name of the policy document to get. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes
RoleName	Name of the role associated with the policy. Type: String Length constraints: Minimum length of 1. Maximum length of 64.	Yes

Response Elements

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

Name	Description
`PolicyDocument`	The policy document. Type: String
`PolicyName`	The name of the policy. Type: String
`RoleName`	The role the policy is associated with. Type: String

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
`NoSuchEntity`	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=GetRolePolicy
&PolicyName=S3AccessPolicy
&RoleName=S3Access
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<GetRolePolicyResponse xmlns="https://iam.amazonaws.com/doc/2010-05-08/">
  <GetRolePolicyResult>
    <PolicyName>S3AccessPolicy</PolicyName>
    <RoleName>S3Access</RoleName>
    <PolicyDocument>{"Version":"2008-10-17","Statement":[{"Effect":"Allow","Ac
tion":["s3:*"],"Resource":["*"]}]}</PolicyDocument>
  </GetRolePolicyResult>
  <ResponseMetadata>
    <RequestId>7e7cd8bc-99ef-11e1-a4c3-27EXAMPLE804</RequestId>
  </ResponseMetadata>
</GetRolePolicyResponse>
```

GetServerCertificate

Description

Retrieves information about the specified server certificate.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
ServerCertificateName	The name of the server certificate you want to retrieve information about. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes

Response Elements

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

Name	Description
`ServerCertificate`	Information about the server certificate. Type: ServerCertificate (p. 172)

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
`NoSuchEntity`	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=GetServerCertificate
&ServerCertificateName=ProdServerCert
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<GetServerCertificateResponse>
  <GetServerCertificateResult>
    <ServerCertificate>
      <ServerCertificateMetadata>
        <ServerCertificateName>ProdServerCert</ServerCertificateName>
        <Path>/company/servercerts/</Path>
        <Arn>arn:aws:iam::123456789012:server-certificate/company/server
certs/ProdServerCert</Arn>
        <UploadDate>2010-05-08T01:02:03.004Z</UploadDate>
        <ServerCertificateId>ASCACKCEVSQ6C2EXAMPLE</ServerCertificateId>
      </ServerCertificateMetadata>
      <CertificateBody>-----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-----
      </CertificateBody>
    </ServerCertificate>
  </GetServerCertificateResult>
  <ResponseMetadata>
    <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
  </ResponseMetadata>
</GetServerCertificateResponse>
```

GetUser

Description

Retrieves information about the specified user, including the user's path, GUID, and ARN.

If you do not specify a user name, IAM determines the user name implicitly based on the AWS Access Key ID signing the request.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
UserName	Name of the user to get information about. This parameter is optional. If it is not included, it defaults to the user making the request. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	No

Response Elements

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

Name	Description
`User`	Information about the user. Type: User (p. 175)

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
`NoSuchEntity`	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=GetUser
&UserName=Bob
```

```
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<GetUserResponse>
   <GetUserResult>
      <User>
         <Path>/division_abc/subdivision_xyz/</Path>
         <UserName>Bob</UserName>
         <UserId>AIDACKCEVSQ6C2EXAMPLE</UserId>
         <Arn>
         arn:aws:iam::123456789012:user/division_abc/subdivision_xyz/Bob
         </Arn>
      </User>
   </GetUserResult>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</GetUserResponse>
```

GetUserPolicy

Description

Retrieves the specified policy document for the specified user. The returned policy is URL-encoded according to RFC 3986. For more information about RFC 3986, go to http://www.faqs.org/rfcs/rfc3986.html.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
PolicyName	Name of the policy document to get. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes
UserName	Name of the user who the policy is associated with. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes

Response Elements

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

Name	Description
`PolicyDocument`	The policy document. Type: String
`PolicyName`	The name of the policy. Type: String
`UserName`	The user the policy is associated with. Type: String

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
`NoSuchEntity`	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=GetUserPolicy
&UserName=Bob
&PolicyName=AllAccessPolicy
&AUTHPARAMS
```

Sample Response

```
<GetUserPolicyResponse>
   <GetUserPolicyResult>
      <UserName>Bob</UserName>
      <PolicyName>AllAccessPolicy</PolicyName>
      <PolicyDocument>
      {"Statement":[{"Effect":"Allow","Action":"*","Resource":"*"}]}
      </PolicyDocument>
   </GetUserPolicyResult>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</GetUserPolicyResponse>
```

ListAccessKeys

Description

Returns information about the Access Key IDs associated with the specified user. If there are none, the action returns an empty list.

Although each user is limited to a small number of keys, you can still paginate the results using the `MaxItems` and `Marker` parameters.

If the `UserName` field is not specified, the UserName is determined implicitly based on the AWS Access Key ID used to sign the request. Because this action works for access keys under the AWS account, this API can be used to manage root credentials even if the AWS account has no associated users.

> **Note**
>
> To ensure the security of your AWS account, the secret access key is accessible only during key and user creation.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
Marker	Use this parameter 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 Length constraints: Minimum length of 1. Maximum length of 320.	No
MaxItems	Use this parameter only when paginating results to indicate the maximum number of keys you want in the response. If there are additional keys beyond the maximum you specify, the `IsTruncated` response element is `true`. This parameter is optional. If you do not include it, it defaults to 100. Type: Integer	No
UserName	Name of the user. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	No

Response Elements

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

Name	Description
AccessKeyMetadata	A list of access key metadata. Type: AccessKeyMetadata (p. 152) list

Name	Description
IsTruncated	A flag that indicates whether there are more keys to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more keys in the list. Type: Boolean
Marker	If `IsTruncated` is `true`, this element is present and contains the value to use for the `Marker` parameter in a subsequent pagination request. Type: String

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=ListAccessKeys
&UserName=Bob
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<ListAccessKeysResponse>
   <ListAccessKeysResult>
      <UserName>Bob</UserName>
      <AccessKeyMetadata>
         <member>
            <UserName>Bob</UserName>
            <AccessKeyId>AKIAIOSFODNN7EXAMPLE</AccessKeyId>
            <Status>Active</Status>
         </member>
         <member>
            <UserName>Bob</UserName>
            <AccessKeyId>AKIAI44QH8DHBEXAMPLE</AccessKeyId>
            <Status>Inactive</Status>
         </member>
      </AccessKeyMetadata>
```

```
      <IsTruncated>false</IsTruncated>
   </ListAccessKeysResult>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</ListAccessKeysResponse>
```

ListAccountAliases

Description

Lists the account aliases associated with the account. For information about using an AWS account alias, see Using an Alias for Your AWS Account ID in *Using AWS Identity and Access Management*.

You can paginate the results using the `MaxItems` and `Marker` parameters.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
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 Length constraints: Minimum length of 1. Maximum length of 320.	No
MaxItems	Use this only when paginating results to indicate the maximum number of account aliases you want in the response. If there are additional account aliases beyond the maximum you specify, the `IsTruncated` response element is `true`. This parameter is optional. If you do not include it, it defaults to 100. Type: Integer	No

Response Elements

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

Name	Description
AccountAliases	A list of aliases associated with the account. Type: String list
IsTruncated	A flag that indicates whether there are more account aliases to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more account aliases in the list. Type: Boolean
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

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=ListAccountAliases
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<ListAccountAliasesResponse>
  <ListAccountAliasesResult>
    <IsTruncated>false</IsTruncated>
    <AccountAliases>
      <member>foocorporation</member>
    </AccountAliases>
  </ListAccountAliasesResult>
  <ResponseMetadata>
    <RequestId>c5a076e9-f1b0-11df-8fbe-45274EXAMPLE</RequestId>
  </ResponseMetadata>
</ListAccountAliasesResponse>
```

ListGroupPolicies

Description

Lists the names of the policies associated with the specified group. If there are none, the action returns an empty list.

You can paginate the results using the `MaxItems` and `Marker` parameters.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
GroupName	The name of the group to list policies for. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes
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 Length constraints: Minimum length of 1. Maximum length of 320.	No
MaxItems	Use this only when paginating results to indicate the maximum number of policy names you want in the response. If there are additional policy names beyond the maximum you specify, the `IsTruncated` response element is `true`. This parameter is optional. If you do not include it, it defaults to 100. Type: Integer	No

Response Elements

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

Name	Description
IsTruncated	A flag that indicates whether there are more policy names to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more policy names in the list. Type: Boolean
Marker	If `IsTruncated` is `true`, this element is present and contains the value to use for the `Marker` parameter in a subsequent pagination request. Type: String
PolicyNames	A list of policy names. Type: String list

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=ListGroupPolicies
&GroupName=Admins
&AUTHPARAMS
```

Sample Response

```
<ListGroupPoliciesResponse>
   <ListGroupPoliciesResult>
      <PolicyNames>
         <member>AdminRoot</member>
         <member>KeyPolicy</member>
      </PolicyNames>
      <IsTruncated>false</IsTruncated>
   </ListGroupPoliciesResult>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</ListGroupPoliciesResponse>
```

ListGroups

Description

Lists the groups that have the specified path prefix.

You can paginate the results using the `MaxItems` and `Marker` parameters.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
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 Length constraints: Minimum length of 1. Maximum length of 320.	No
MaxItems	Use this only when paginating results to indicate the maximum number of groups you want in the response. If there are additional groups beyond the maximum you specify, the `IsTruncated` response element is `true`. This parameter is optional. If you do not include it, it defaults to 100. Type: Integer	No
PathPrefix	The path prefix for filtering the results. For example: `/division_abc/subdivision_xyz/`, which would get all groups whose path starts with `/division_abc/subdivision_xyz/`. This parameter is optional. If it is not included, it defaults to a slash (/), listing all groups. Type: String Length constraints: Minimum length of 1. Maximum length of 512.	No

Response Elements

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

Name	Description
Groups	A list of groups. Type: Group (p. 160) list
IsTruncated	A flag that indicates whether there are more groups to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more groups in the list. Type: Boolean
Marker	If `IsTruncated` is `true`, this element is present and contains the value to use for the `Marker` parameter in a subsequent pagination request. Type: String

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=ListGroups
&PathPrefix=/division_abc/subdivision_xyz/
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<ListGroupsResponse>
   <ListGroupsResult>
      <Groups>
         <member>
            <Path>/division_abc/subdivision_xyz/</Path>
            <GroupName>Admins</GroupName>
            <GroupId>AGPACKCEVSQ6C2EXAMPLE</GroupId>
            <Arn>arn:aws:iam::123456789012:group/Admins</Arn>
         </member>
         <member>
            <Path>/division_abc/subdivision_xyz/product_1234/engineering/
            </Path>
            <GroupName>Test</GroupName>
            <GroupId>AGP2MAB8DPLSRHEXAMPLE</GroupId>
            <Arn>arn:aws:iam::123456789012:group
            /division_abc/subdivision_xyz/product_1234/engineering/Test</Arn>
         </member>
         <member>
            <Path>/division_abc/subdivision_xyz/product_1234/</Path>
            <GroupName>Managers</GroupName>
            <GroupId>AGPIODR4TAW7CSEXAMPLE</GroupId>
            <Arn>arn:aws:iam::123456789012
            :group/division_abc/subdivision_xyz/product_1234/Managers</Arn>
         </member>
      </Groups>
      <IsTruncated>false</IsTruncated>
   </ListGroupsResult>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</ListGroupsResponse>
```

ListGroupsForUser

Description

Lists the groups the specified user belongs to.

You can paginate the results using the `MaxItems` and `Marker` parameters.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
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 Length constraints: Minimum length of 1. Maximum length of 320.	No
MaxItems	Use this only when paginating results to indicate the maximum number of groups you want in the response. If there are additional groups beyond the maximum you specify, the `IsTruncated` response element is `true`. This parameter is optional. If you do not include it, it defaults to 100. Type: Integer	No
UserName	The name of the user to list groups for. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes

Response Elements

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

Name	Description
Groups	A list of groups. Type: Group (p. 160) list
IsTruncated	A flag that indicates whether there are more groups to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more groups in the list. Type: Boolean
Marker	If `IsTruncated` is `true`, this element is present and contains the value to use for the `Marker` parameter in a subsequent pagination request. Type: String

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=ListGroupsForUser
&UserName=Bob
&AUTHPARAMS
```

Sample Response

```
<ListGroupsForUserResponse>
   <ListGroupsForUserResult>
      <Groups>
         <member>
            <Path>/</Path>
            <GroupName>Admins</GroupName>
            <GroupId>AGPACKCEVSQ6C2EXAMPLE</GroupId>
            <Arn>arn:aws:iam::123456789012:group/Admins</Arn>
         </member>
      </Groups>
      <IsTruncated>false</IsTruncated>
   </ListGroupsForUserResult>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</ListGroupsForUserResponse>
```

ListInstanceProfiles

Description

Lists the instance profiles that have the specified path prefix. If there are none, the action returns an empty list. For more information about instance profiles, go to About Instance Profiles.

You can paginate the results using the `MaxItems` and `Marker` parameters.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
Marker	Use this parameter 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 Length constraints: Minimum length of 1. Maximum length of 320.	No
MaxItems	Use this parameter only when paginating results to indicate the maximum number of user names you want in the response. If there are additional user names beyond the maximum you specify, the `IsTruncated` response element is `true`. This parameter is optional. If you do not include it, it defaults to 100. Type: Integer	No
PathPrefix	The path prefix for filtering the results. For example: `/application_abc/component_xyz/`, which would get all instance profiles whose path starts with `/application_abc/component_xyz/`. This parameter is optional. If it is not included, it defaults to a slash (/), listing all instance profiles. Type: String Length constraints: Minimum length of 1. Maximum length of 512.	No

Response Elements

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

Name	Description
InstanceProfiles	A list of instance profiles. Type: InstanceProfile (p. 161) list
IsTruncated	A flag that indicates whether there are more instance profiles to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more instance profiles in the list. Type: Boolean
Marker	If `IsTruncated` is `true`, this element is present and contains the value to use for the `Marker` parameter in a subsequent pagination request. Type: String

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=ListInstanceProfiles
&MaxItems=100
&PathPrefix=/application_abc/
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<ListInstanceProfilesResponse xmlns="https://iam.amazonaws.com/doc/2010-05-08/">

  <ListInstanceProfilesResult>
    <IsTruncated>false</IsTruncated>
    <InstanceProfiles>
      <member>
        <Id>AIPACIFN4OZXG7EXAMPLE</Id>
        <Roles/>
        <InstanceProfileName>Database</InstanceProfileName>
        <Path>/application_abc/component_xyz/</Path>
       <Arn>arn:aws:iam::123456789012:instance-profile/application_abc/compon
ent_xyz/Database</Arn>
        <CreateDate>2012-05-09T16:27:03Z</CreateDate>
      </member>
      <member>
        <Id>AIPACZLSXM2EYYEXAMPLE</Id>
        <Roles/>
        <InstanceProfileName>Webserver</InstanceProfileName>
        <Path>/application_abc/component_xyz/</Path>
       <Arn>arn:aws:iam::123456789012:instance-profile/application_abc/compon
ent_xyz/Webserver</Arn>
        <CreateDate>2012-05-09T16:27:11Z</CreateDate>
      </member>
    </InstanceProfiles>
  </ListInstanceProfilesResult>
  <ResponseMetadata>
    <RequestId>fd74fa8d-99f3-11e1-a4c3-27EXAMPLE804</RequestId>
  </ResponseMetadata>
</ListInstanceProfilesResponse>
```

ListInstanceProfilesForRole

Description

Lists the instance profiles that have the specified associated role. If there are none, the action returns an empty list. For more information about instance profiles, go to About Instance Profiles.

You can paginate the results using the `MaxItems` and `Marker` parameters.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
Marker	Use this parameter 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 Length constraints: Minimum length of 1. Maximum length of 320.	No
MaxItems	Use this parameter only when paginating results to indicate the maximum number of user names you want in the response. If there are additional user names beyond the maximum you specify, the `IsTruncated` response element is `true`. This parameter is optional. If you do not include it, it defaults to 100. Type: Integer	No
RoleName	The name of the role to list instance profiles for. Type: String Length constraints: Minimum length of 1. Maximum length of 64.	Yes

Response Elements

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

Name	Description
InstanceProfiles	A list of instance profiles. Type: InstanceProfile (p. 161) list
IsTruncated	A flag that indicates whether there are more instance profiles to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more instance profiles in the list. Type: Boolean
Marker	If `IsTruncated` is `true`, this element is present and contains the value to use for the `Marker` parameter in a subsequent pagination request. Type: String

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=ListInstanceProfilesForRole
&MaxItems=100
&RoleName=S3Access
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<ListInstanceProfilesForRoleResponse xmlns="https://iam.amazonaws.com/doc/2010-
05-08/">
  <ListInstanceProfilesForRoleResult>
    <IsTruncated>false</IsTruncated>
    <InstanceProfiles>
      <member>
        <Id>AIPACZLS2EYYXMEXAMPLE</Id>
        <Roles>
          <member>
            <Path>/application_abc/component_xyz/</Path>
            <Arn>arn:aws:iam::123456789012:role/application_abc/compon
ent_xyz/S3Access</Arn>
            <RoleName>S3Access</RoleName>
          <AssumeRolePolicyDocument>{"Version":"2008-10-17","Statement":[{"Ef
fect":"Allow","Principal":{"Service":["ec2.amazonaws.com"]},"Action":["sts:As
sumeRole"]}]}</AssumeRolePolicyDocument>
            <CreateDate>2012-05-09T15:45:35Z</CreateDate>
            <RoleId>AROACVSVTSZYK3EXAMPLE</RoleId>
          </member>
        </Roles>
        <InstanceProfileName>Webserver</InstanceProfileName>
        <Path>/application_abc/component_xyz/</Path>
      <Arn>arn:aws:iam::123456789012:instance-profile/application_abc/compon
ent_xyz/Webserver</Arn>
        <CreateDate>2012-05-09T16:27:11Z</CreateDate>
      </member>
    </InstanceProfiles>
  </ListInstanceProfilesForRoleResult>
  <ResponseMetadata>
    <RequestId>6a8c3992-99f4-11e1-a4c3-27EXAMPLE804</RequestId>
```

```
  </ResponseMetadata>
</ListInstanceProfilesForRoleResponse>
```

ListMFADevices

Description

Lists the MFA devices. If the request includes the user name, then this action lists all the MFA devices 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 signing the request.

You can paginate the results using the `MaxItems` and `Marker` parameters.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
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 Length constraints: Minimum length of 1. Maximum length of 320.	No
MaxItems	Use this only when paginating results to indicate the maximum number of MFA devices you want in the response. If there are additional MFA devices beyond the maximum you specify, the `IsTruncated` response element is `true`. This parameter is optional. If you do not include it, it defaults to 100. Type: Integer	No
UserName	Name of the user whose MFA devices you want to list. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	No

Response Elements

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

Name	Description
IsTruncated	A flag that indicates whether there are more MFA devices to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more MFA devices in the list. Type: Boolean
MFADevices	A list of MFA devices. Type: MFADevice (p. 170) list
Marker	If `IsTruncated` is `true`, this element is present and contains the value to use for the `Marker` parameter in a subsequent pagination request. Type: String

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=ListMFADevices
&UserName=Bob
&AUTHPARAMS
```

Sample Response

```
<ListMFADevicesResponse>
   <ListMFADevicesResult>
      <MFADevices>
         <member>
            <UserName>Bob</UserName>
            <SerialNumber>R1234</SerialNumber>
         </member>
      </MFADevices>
      <IsTruncated>false</IsTruncated>
   </ListMFADevicesResult>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</ListMFADevicesResponse>
```

ListRolePolicies

Description

Lists the names of the policies associated with the specified role. If there are none, the action returns an empty list.

You can paginate the results using the `MaxItems` and `Marker` parameters.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
Marker	Use this parameter 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 Length constraints: Minimum length of 1. Maximum length of 320.	No
MaxItems	Use this parameter only when paginating results to indicate the maximum number of user names you want in the response. If there are additional user names beyond the maximum you specify, the `IsTruncated` response element is `true`. This parameter is optional. If you do not include it, it defaults to 100. Type: Integer	No
RoleName	The name of the role to list policies for. Type: String Length constraints: Minimum length of 1. Maximum length of 64.	Yes

Response Elements

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

Name	Description
IsTruncated	A flag that indicates whether there are more policy names to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more policy names in the list. Type: Boolean
Marker	If `IsTruncated` is `true`, this element is present and contains the value to use for the `Marker` parameter in a subsequent pagination request. Type: String
PolicyNames	A list of policy names. Type: String list

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=ListRolePolicies
&MaxItems=100
&RoleName=S3Access
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<ListRolePoliciesResponse xmlns="https://iam.amazonaws.com/doc/2010-05-08/">
  <ListRolePoliciesResult>
    <PolicyNames>
      <member>CloudwatchPutMetricPolicy</member>
      <member>S3AccessPolicy</member>
    </PolicyNames>
    <IsTruncated>false</IsTruncated>
  </ListRolePoliciesResult>
  <ResponseMetadata>
    <RequestId>8c7e1816-99f0-11e1-a4c3-27EXAMPLE804</RequestId>
  </ResponseMetadata>
</ListRolePoliciesResponse>
```

ListRoles

Description

Lists the roles that have the specified path prefix. If there are none, the action returns an empty list. For more information about roles, go to Working with Roles.

You can paginate the results using the `MaxItems` and `Marker` parameters.

The returned policy is URL-encoded according to RFC 3986. For more information about RFC 3986, go to http://www.faqs.org/rfcs/rfc3986.html.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
Marker	Use this parameter 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 Length constraints: Minimum length of 1. Maximum length of 320.	No
MaxItems	Use this parameter only when paginating results to indicate the maximum number of user names you want in the response. If there are additional user names beyond the maximum you specify, the `IsTruncated` response element is `true`. This parameter is optional. If you do not include it, it defaults to 100. Type: Integer	No
PathPrefix	The path prefix for filtering the results. For example: `/application_abc/component_xyz/`, which would get all roles whose path starts with `/application_abc/component_xyz/`. This parameter is optional. If it is not included, it defaults to a slash (/), listing all roles. Type: String Length constraints: Minimum length of 1. Maximum length of 512.	No

Response Elements

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

Name	Description
IsTruncated	A flag that indicates whether there are more roles to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more roles in the list. Type: Boolean
Marker	If `IsTruncated` is `true`, this element is present and contains the value to use for the `Marker` parameter in a subsequent pagination request. Type: String

Name	Description
Roles	A list of roles. Type: Role (p. 171) list

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=ListRoles
&MaxItems=100
&PathPrefix=/application_abc/
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<ListRolesResponse xmlns="https://iam.amazonaws.com/doc/2010-05-08/">
  <ListRolesResult>
    <IsTruncated>false</IsTruncated>
    <Roles>
      <member>
        <Path>/application_abc/component_xyz/</Path>
        <Arn>arn:aws:iam::123456789012:role/application_abc/component_xyz/S3Ac
cess</Arn>
        <RoleName>S3Access</RoleName>
        <AssumeRolePolicyDocument>{"Version":"2008-10-17","Statement":[{"Ef
fect":"Allow","Principal":{"Service":["ec2.amazonaws.com"]},"Action":["sts:As
sumeRole"]}]}</AssumeRolePolicyDocument>
        <CreateDate>2012-05-09T15:45:35Z</CreateDate>
        <RoleId>AROACVSVTSZYEXAMPLEYK</RoleId>
      </member>
      <member>
        <Path>/application_abc/component_xyz/</Path>
        <Arn>arn:aws:iam::123456789012:role/application_abc/component_xyz/SDBAc
cess</Arn>
        <RoleName>SDBAccess</RoleName>
        <AssumeRolePolicyDocument>{"Version":"2008-10-17","Statement":[{"Ef
fect":"Allow","Principal":{"Service":["ec2.amazonaws.com"]},"Action":["sts:As
sumeRole"]}]}</AssumeRolePolicyDocument>
        <CreateDate>2012-05-09T15:45:45Z</CreateDate>
        <RoleId>AROAC2ICXG32EXAMPLEWK</RoleId>
      </member>
    </Roles>
  </ListRolesResult>
  <ResponseMetadata>
    <RequestId>20f7279f-99ee-11e1-a4c3-27EXAMPLE804</RequestId>
  </ResponseMetadata>
```

```
</ListRolesResponse>
```

ListServerCertificates

Description

Lists the server certificates that have the specified path prefix. If none exist, the action returns an empty list.

You can paginate the results using the `MaxItems` and `Marker` parameters.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
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 Length constraints: Minimum length of 1. Maximum length of 320.	No
MaxItems	Use this only when paginating results to indicate the maximum number of server certificates you want in the response. If there are additional server certificates beyond the maximum you specify, the `IsTruncated` response element will be set to `true`. This parameter is optional. If you do not include it, it defaults to 100. Type: Integer	No
PathPrefix	The path prefix for filtering the results. For example: `/company/servercerts` would get all server certificates for which the path starts with `/company/servercerts`. This parameter is optional. If it is not included, it defaults to a slash (/), listing all server certificates. Type: String Length constraints: Minimum length of 1. Maximum length of 512.	No

Response Elements

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

Name	Description
IsTruncated	A flag that indicates whether there are more server certificates to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more server certificates in the list. Type: Boolean
Marker	If `IsTruncated` is `true`, this element is present and contains the value to use for the `Marker` parameter in a subsequent pagination request. Type: String

Name	Description
`ServerCertificateMetadataList`	A list of server certificates. Type: ServerCertificateMetadata (p. 173) list

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=ListServerCertificates
&PathPrefix=/company/servercerts
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<ListServerCertificatesResponse>
  <ListServerCertificatesResult>
    <IsTruncated>false</IsTruncated>
    <ServerCertificateMetadataList>
      <member>
        <ServerCertificateMetadata>
          <ServerCertificateName>ProdServerCert</ServerCertificateName>
          <Path>/company/servercerts/</Path>
          <Arn>arn:aws:iam::123456789012:server-certificate/company/server
certs/ProdServerCert</Arn>
          <UploadDate>2010-05-08T01:02:03.004Z</UploadDate>
          <ServerCertificateId>ASCACKCEVSQ6CEXAMPLE1</ServerCertificateId>
        </ServerCertificateMetadata>
      </member>
      <member>
        <ServerCertificateMetadata>
          <ServerCertificateName>BetaServerCert</ServerCertificateName>
          <Path>/company/servercerts/</Path>
          <Arn>arn:aws:iam::123456789012:server-certificate/company/server
certs/BetaServerCert</Arn>
          <UploadDate>2010-05-08T02:03:01.004Z</UploadDate>
          <ServerCertificateId>ASCACKCEVSQ6CEXAMPLE2</ServerCertificateId>
        </ServerCertificateMetadata>
      </member>
      <member>
        <ServerCertificateMetadata>
          <ServerCertificateName>TestServerCert</ServerCertificateName>
          <Path>/company/servercerts/</Path>
          <Arn>arn:aws:iam::123456789012:server-certificate/company/server
certs/TestServerCert</Arn>
          <UploadDate>2010-05-08T03:01:02.004Z</UploadDate>
          <ServerCertificateId>ASCACKCEVSQ6CEXAMPLE3</ServerCertificateId>
        </ServerCertificateMetadata>
```

```
        </member>
      </ServerCertificateMetadataList>
  </ListServerCertificatesResult>
  <ResponseMetadata>
    <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
  </ResponseMetadata>
</ListServerCertificatesResponse>
```

ListSigningCertificates

Description

Returns information about the signing certificates associated with the specified user. If there are none, the action returns an empty list.

Although each user is limited to a small number of signing certificates, you can still paginate the results using the `MaxItems` and `Marker` parameters.

If the `UserName` field is not specified, the user name is determined implicitly based on the AWS Access Key ID used to sign the request. Because this action works for access keys under the AWS account, this API can be used to manage root credentials even if the AWS account has no associated users.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
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 Length constraints: Minimum length of 1. Maximum length of 320.	No
MaxItems	Use this only when paginating results to indicate the maximum number of certificate IDs you want in the response. If there are additional certificate IDs beyond the maximum you specify, the `IsTruncated` response element is `true`. This parameter is optional. If you do not include it, it defaults to 100. Type: Integer	No
UserName	The name of the user. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	No

Response Elements

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

Name	Description
Certificates	A list of the user's signing certificate information. Type: SigningCertificate (p. 174) list
IsTruncated	A flag that indicates whether there are more certificate IDs to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more certificates in the list. Type: Boolean
Marker	If `IsTruncated` is `true`, this element is present and contains the value to use for the `Marker` parameter in a subsequent pagination request. Type: String

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=ListSigningCertificates
&UserName=Bob
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<ListSigningCertificatesResponse>
   <ListSigningCertificatesResult>
      <UserName>Bob</UserName>
      <Certificates>
         <member>
            <UserName>Bob</UserName>
            <CertificateId>TA7SMP42TDN5Z26OBPJE7EXAMPLE</CertificateId>
            <CertificateBody>-----BEGIN CERTIFICATE-----
   MIICdzCCAeCgAwIBAgIGANc+Ha2wMA0GCSqGSIb3DQEBBQUAMFMxCzAJBgNVBAYT
   AlVTMRMwEQYDVQQKEwpBbWF6b24uY29tMQwwCgYDVQQLEwNBV1MxITAfBgNVBAMT
   GEFXUyBMaW1pdGVkLUFzc3VyYW5jZSBDQTAeFw0wOTAyMDQxNzE5MjdaFw0xMDAy
   MDQxNzE5MjdaMFIxCzAJBgNVBAYTAlVTMRMwEQYDVQQKEwpBbWF6b24uY29tMRcw
   FQYDVQQLEw5BV1MtRGV2ZWxvcGVyczEVMBMGA1UEAxMMNTdkxNDl0c3ZwYjRtMIGf
   MA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCpB/vsOwmT/O0td1RqzKjttSBaPjbr
   dqwNe9BrOyB08fw2+Ch5oonZYXfGUrT6mkYXH5fQot9HvASrzAKHO596FdJA6DmL
   ywdWelOggk7zFSXO1Xv+3vPrJtaYxYo3eRIp7w80PMkiOv6M0XK8ubcTouODeJbf
   suDqcLnLDxwsvwIDAQABo1cwVTAOBgNVHQ8BAf8EBAMCBaAwFgYDVR0lAQH/BAww
   CgYIKwYBBQUHAwIwDAYDVR0TAQH/BAIwADAdBgNVHQ4EFgQULGNaBphBumaKbDRK
   CAi0mH8B3mowDQYJKoZIhvcNAQEFBQADgYEAuKxhkXaCLGcqDuweKtO/AEw9ZePH
   wr0XqsaIK2HZboqruebXEGsojK4Ks0WzwgrEynuHJwTn760xe39rSqXWIOGrOBaX
   wFpWHVjTFMKk+tSDG1lssLHyYWWdFFU4AnejRGORJYNaRHgVTKjHphc5jEhHm0BX
   AEaHzTpmEXAMPLE=
   -----END CERTIFICATE-----</CertificateBody>
            <Status>Active</Status>
         </member>
      </Certificates>
      <IsTruncated>false</IsTruncated>
```

```
    </ListSigningCertificatesResult>
    <ResponseMetadata>
        <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
    </ResponseMetadata>
</ListSigningCertificatesResponse>
```

ListUserPolicies

Description

Lists the names of the policies associated with the specified user. If there are none, the action returns an empty list.

You can paginate the results using the `MaxItems` and `Marker` parameters.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
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 Length constraints: Minimum length of 1. Maximum length of 320.	No
MaxItems	Use this only when paginating results to indicate the maximum number of policy names you want in the response. If there are additional policy names beyond the maximum you specify, the `IsTruncated` response element is `true`. This parameter is optional. If you do not include it, it defaults to 100. Type: Integer	No
UserName	The name of the user to list policies for. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes

Response Elements

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

Name	Description
IsTruncated	A flag that indicates whether there are more policy names to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more policy names in the list. Type: Boolean
Marker	If `IsTruncated` is `true`, this element is present and contains the value to use for the `Marker` parameter in a subsequent pagination request. Type: String
PolicyNames	A list of policy names. Type: String list

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=ListUserPolicies
&UserName=Bob
&AUTHPARAMS
```

Sample Response

```
<ListUserPoliciesResponse>
   <ListUserPoliciesResult>
      <PolicyNames>
         <member>AllAccessPolicy</member>
         <member>KeyPolicy</member>
      </PolicyNames>
      <IsTruncated>false</IsTruncated>
   </ListUserPoliciesResult>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</ListUserPoliciesResponse>
```

ListUsers

Description

Lists the users 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` parameters.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
Marker	Use this parameter 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 Length constraints: Minimum length of 1. Maximum length of 320.	No
MaxItems	Use this parameter only when paginating results to indicate the maximum number of user names you want in the response. If there are additional user names beyond the maximum you specify, the `IsTruncated` response element is `true`. This parameter is optional. If you do not include it, it defaults to 100. Type: Integer	No
PathPrefix	The path prefix for filtering the results. For example: `/division_abc/subdivision_xyz/`, which would get all user names whose path starts with `/division_abc/subdivision_xyz/`. This parameter is optional. If it is not included, it defaults to a slash (/), listing all user names. Type: String Length constraints: Minimum length of 1. Maximum length of 512.	No

Response Elements

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

Name	Description
IsTruncated	A flag that indicates whether there are more user names to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more users in the list. Type: Boolean
Marker	If `IsTruncated` is `true`, this element is present and contains the value to use for the `Marker` parameter in a subsequent pagination request. Type: String
Users	A list of users. Type: User (p. 175) list

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=ListUsers
&PathPrefix=/division_abc/subdivision_xyz/product_1234/engineering/
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<ListUsersResponse>
   <ListUsersResult>
      <Users>
         <member>
             <Path>/division_abc/subdivision_xyz/engineering/</Path>
             <UserName>Andrew</UserName>
             <UserId>AID2MAB8DPLSRHEXAMPLE</UserId>
             <Arn>arn:aws:iam::123456789012:user
             /division_abc/subdivision_xyz/engineering/Andrew</Arn>
         </member>
         <member>
             <Path>/division_abc/subdivision_xyz/engineering/</Path>
             <UserName>Jackie</UserName>
             <UserId>AIDIODR4TAW7CSEXAMPLE</UserId>
             <Arn>arn:aws:iam::123456789012:user
             /division_abc/subdivision_xyz/engineering/Jackie</Arn>
         </member>
      </Users>
      <IsTruncated>false</IsTruncated>
   </ListUsersResult>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</ListUsersResponse>
```

ListVirtualMFADevices

Description

Lists the virtual MFA devices under the AWS account by assignment status. If you do not specify an assignment status, the action 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` parameters.

Request Parameters

For information about the common parameters that all actions use, see .

Name	Description	Required
AssignmentStatus	The status (unassigned or assigned) of the devices to list. If you do not specify an `AssignmentStatus`, the action defaults to `Any` which lists both assigned and unassigned virtual MFA devices. Type: String Valid Values: `Assigned` | `Unassigned` | `Any`	No
Marker	Use this parameter 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 Length constraints: Minimum length of 1. Maximum length of 320.	No
MaxItems	Use this parameter only when paginating results to indicate the maximum number of user names you want in the response. If there are additional user names beyond the maximum you specify, the `IsTruncated` response element is `true`. This parameter is optional. If you do not include it, it defaults to 100. Type: Integer	No

Response Elements

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

Name	Description
IsTruncated	A flag that indicates whether there are more items to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more items the list. Type: Boolean
Marker	If `IsTruncated` is `true`, this element is present and contains the value to use for the `Marker` parameter in a subsequent pagination request. Type: String
VirtualMFADevices	Type: VirtualMFADevice (p. 176) list

Examples

Sample Request

```
<!-- This example shows the request where
     the AssignmentStatus is Any -->

https://iam.amazonaws.com/
?Action=ListVirtualMFADevices
&AssignmentStatus=Any
&AUTHPARAMS
```

Sample Response

```
<!-- The action returns all three virtual MFA devices
     associated with the account: the first device is
     unassigned, the second is assigned to the root
     account, and the third is assigned to a user
     named ExampleUser under the account. -->

<ListVirtualMFADevicesResponse>
  <ListVirtualMFADevicesResult>
    <IsTruncated>false</IsTruncated>
    <VirtualMFADevices>
      <member>
        <SerialNumber>
        arn:aws:iam::123456789012:mfa/MFAdeviceName
        </SerialNumber>
      </member>
      <member>
        <SerialNumber>
        arn:aws:iam::123456789012:mfa/RootMFAdeviceName
        </SerialNumber>
        <EnableDate>2011-10-20T20:49:03Z</EnableDate>
        <User>
          <UserId>123456789012</UserId>
          <Arn>arn:aws:iam::123456789012:root</Arn>
          <CreateDate>2009-10-13T22:00:36Z</CreateDate>
        </User>
      </member>
      <member>
        <SerialNumber>
        arn:aws:iam:::mfa/ExampleUserMFAdeviceName
        </SerialNumber>
        <EnableDate>2011-10-31T20:45:02Z</EnableDate>
        <User>
          <UserId>AIDEXAMPLE4EXAMPLEXYZ</UserId>
          <Path>/</Path>
          <UserName>ExampleUser</UserName>
          <Arn>arn:aws:iam::111122223333:user/ExampleUser</Arn>
          <CreateDate>2011-07-01T17:23:07Z</CreateDate>
        </User>
      </member>
```

```
        </VirtualMFADevices>
  </ListVirtualMFADevicesResult>
  <ResponseMetadata>
    <RequestId>b61ce1b1-0401-11e1-b2f8-2dEXAMPLEbfc</RequestId>
  </ResponseMetadata>
</ListVirtualMFADevicesResponse>
```

PutGroupPolicy

Description

Adds (or updates) a policy document associated with the specified group. For information about policies, refer to Overview of Policies in *Using AWS Identity and Access Management*.

For information about limits on the number of policies you can associate with a group, see Limitations on IAM Entities in *Using AWS Identity and Access Management*.

> **Note**
>
> Because policy documents can be large, you should use POST rather than GET when calling `PutGroupPolicy`. For information about setting up signatures and authorization through the API, go to Signing AWS API Requests in the *AWS General Reference*. For general information about using the Query API with IAM, go to Making Query Requests in *Using IAM*.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
GroupName	Name of the group to associate the policy with. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes
PolicyDocument	The policy document. Type: String Length constraints: Minimum length of 1. Maximum length of 131072.	Yes
PolicyName	Name of the policy document. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
LimitExceeded	The request was rejected because it attempted to create resources beyond the current AWS account limits. The error message describes the limit exceeded.	409
MalformedPolicyDocument	The request was rejected because the policy document was malformed. The error message describes the specific error.	400
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=PutGroupPolicy
&GroupName=Admins
&PolicyName=AdminRoot
&PolicyDocument={"Statement":[{"Effect":"Allow","Action":"*","Resource":"*"}]}
&AUTHPARAMS
```

Sample Response

```
<PutGroupPolicyResponse>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</PutGroupPolicyResponse>
```

PutRolePolicy

Description

Adds (or updates) a policy document associated with the specified role. For information about policies, go to Overview of Policies in *Using AWS Identity and Access Management*.

For information about limits on the policies you can associate with a role, see Limitations on IAM Entities in *Using AWS Identity and Access Management*.

> **Note**
>
> Because policy documents can be large, you should use POST rather than GET when calling `PutRolePolicy`. For information about setting up signatures and authorization through the API, go to Signing AWS API Requests in the *AWS General Reference*. For general information about using the Query API with IAM, go to Making Query Requests in *Using IAM*.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
PolicyDocument	The policy document. Type: String Length constraints: Minimum length of 1. Maximum length of 131072.	Yes
PolicyName	Name of the policy document. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes
RoleName	Name of the role to associate the policy with. Type: String Length constraints: Minimum length of 1. Maximum length of 64.	Yes

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
LimitExceeded	The request was rejected because it attempted to create resources beyond the current AWS account limits. The error message describes the limit exceeded.	409
MalformedPolicyDocument	The request was rejected because the policy document was malformed. The error message describes the specific error.	400
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=PutRolePolicy
&RoleName=S3Access
&PolicyName=S3AccessPolicy
&PolicyDocument={"Statement":[{"Effect":"Allow","Action":"s3:*","Resource":"*"}]}
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<PutRolePolicyResponse>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</PutRolePolicyResponse>
```

PutUserPolicy

Description

Adds (or updates) a policy document associated with the specified user. For information about policies, refer to Overview of Policies in *Using AWS Identity and Access Management*.

For information about limits on the number of policies you can associate with a user, see Limitations on IAM Entities in *Using AWS Identity and Access Management*.

> **Note**
>
> Because policy documents can be large, you should use POST rather than GET when calling `PutUserPolicy`. For information about setting up signatures and authorization through the API, go to Signing AWS API Requests in the *AWS General Reference*. For general information about using the Query API with IAM, go to Making Query Requests in *Using IAM*.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
PolicyDocument	The policy document. Type: String Length constraints: Minimum length of 1. Maximum length of 131072.	Yes
PolicyName	Name of the policy document. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes
UserName	Name of the user to associate the policy with. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
LimitExceeded	The request was rejected because it attempted to create resources beyond the current AWS account limits. The error message describes the limit exceeded.	409
MalformedPolicyDocument	The request was rejected because the policy document was malformed. The error message describes the specific error.	400
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=PutUserPolicy
&UserName=Bob
&PolicyName=AllAccessPolicy
&PolicyDocument={"Statement":[{"Effect":"Allow","Action":"*","Resource":"*"}]}
&AUTHPARAMS
```

Sample Response

```
<PutUserPolicyResponse>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</PutUserPolicyResponse>
```

RemoveRoleFromInstanceProfile

Description

Removes the specified role from the specified instance profile.

> ⚠️ **Important**
>
> Make sure you do not have any Amazon EC2 instances running with the role you are about to remove from the instance profile. Removing a role from an instance profile that is associated with a running instance will break any applications running on the instance.

For more information about roles, go to Working with Roles. For more information about instance profiles, go to About Instance Profiles.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
InstanceProfileName	Name of the instance profile to update. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes
RoleName	Name of the role to remove. Type: String Length constraints: Minimum length of 1. Maximum length of 64.	Yes

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=RemoveRoleFromInstanceProfile
&InstanceProfileName=Webserver
```

```
&RoleName=S3Access
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<RemoveRoleFromInstanceProfileResponse xmlns="https://iam.amazonaws.com/doc/2010-
05-08/">
  <ResponseMetadata>
    <RequestId>29f47818-99f5-11e1-a4c3-27EXAMPLE804</RequestId>
  </ResponseMetadata>
</RemoveRoleFromInstanceProfileResponse>
```

RemoveUserFromGroup

Description

Removes the specified user from the specified group.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
GroupName	Name of the group to update. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes
UserName	Name of the user to remove. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=RemoveUserFromGroup
&GroupName=Managers
&UserName=Bob
&AUTHPARAMS
```

Sample Response

```
<RemoveUserFromGroupResponse>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
```

```
</RemoveUserFromGroupResponse>
```

ResyncMFADevice

Description

Synchronizes the specified MFA device with AWS servers.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
AuthenticationCode1	An authentication code emitted by the device. Type: String Length constraints: Minimum length of 6. Maximum length of 6.	Yes
AuthenticationCode2	A subsequent authentication code emitted by the device. Type: String Length constraints: Minimum length of 6. Maximum length of 6.	Yes
SerialNumber	Serial number that uniquely identifies the MFA device. Type: String Length constraints: Minimum length of 9. Maximum length of 256.	Yes
UserName	Name of the user whose MFA device you want to resynchronize. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
InvalidAuthenticationCode	The request was rejected because the authentication code was not recognized. The error message describes the specific error.	403
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=ResyncMFADevice
&UserName=Bob
&SerialNumber=R1234
&AuthenticationCode1=234567
&AuthenticationCode2=987654
&AUTHPARAMS
```

Sample Response

```
<ResyncMFADeviceResponse>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</ResyncMFADeviceResponse>
```

UpdateAccessKey

Description

Changes the status of the specified access key from Active to Inactive, or vice versa. This action can be used to disable a user's key as part of a key rotation work flow.

If the `UserName` field is not specified, the UserName is determined implicitly based on the AWS Access Key ID used to sign the request. Because this action works for access keys under the AWS account, this API can be used to manage root credentials even if the AWS account has no associated users.

For information about rotating keys, see Managing Keys and Certificates in *Using AWS Identity and Access Management.*

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
AccessKeyId	The Access Key ID of the Secret Access Key you want to update. Type: String Length constraints: Minimum length of 16. Maximum length of 32.	Yes
Status	The status you want to assign to the Secret Access Key. `Active` means the key can be used for API calls to AWS, while `Inactive` means the key cannot be used. Type: String Valid Values: `Active` | `Inactive`	Yes
UserName	Name of the user whose key you want to update. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	No

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
```

```
?Action=UpdateAccessKey
&UserName=Bob
&AccessKeyId=AKIAIOSFODNN7EXAMPLE
&Status=Inactive
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<UpdateAccessKeyResponse>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</UpdateAccessKeyResponse>
```

UpdateAccountPasswordPolicy

Description

Updates the password policy settings for the account. For more information about using a password policy, go to Managing an IAM Password Policy.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
AllowUsersToChangePassword	Type: Boolean	No
MinimumPasswordLength	Type: Integer	No
RequireLowercaseCharacters	Type: Boolean	No
RequireNumbers	Type: Boolean	No
RequireSymbols	Type: Boolean	No
RequireUppercaseCharacters	Type: Boolean	No

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
MalformedPolicyDocument	The request was rejected because the policy document was malformed. The error message describes the specific error.	400
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=UpdateAccountPasswordPolicy
&MinimumPasswordLength=9
&RequireSymbols=true
&RequireNumbers=false
&RequireUppercaseCharacters=true
&RequireLowercaseCharacters=true
```

```
&AllowUsersToChangePassword=true
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<UpdateAccountPasswordPolicyResponse>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</UpdateAccountPasswordPolicyResponse>
```

UpdateAssumeRolePolicy

Description

Updates the policy that grants an entity permission to assume a role. Currently, only an Amazon EC2 instance can assume a role. For more information about roles, go to Working with Roles.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
PolicyDocument	The policy that grants an entity permission to assume the role. Type: String Length constraints: Minimum length of 1. Maximum length of 131072.	Yes
RoleName	Name of the role to update. Type: String Length constraints: Minimum length of 1. Maximum length of 64.	Yes

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
MalformedPolicyDocument	The request was rejected because the policy document was malformed. The error message describes the specific error.	400
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=UpdateAssumeRolePolicy
&PolicyDocument={"Version":"2008-10-17","Statement":[{"Effect":"Allow","Princip
al":{"Service":["ec2.amazonaws.com"]},"Action":["sts:AssumeRole"]}]}
&RoleName=S3Access
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<UpdateAssumeRolePolicyResponse xmlns="https://iam.amazonaws.com/doc/2010-05-
08/">
  <ResponseMetadata>
    <RequestId>309c1671-99ed-11e1-a4c3-270EXAMPLE04</RequestId>
  </ResponseMetadata>
</UpdateAssumeRolePolicyResponse>
```

UpdateGroup

Description

Updates the name and/or the path of the specified group.

> ⚠️ **Important**
>
> You should 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.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
GroupName	Name of the group to update. If you're changing the name of the group, this is the original name. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes
NewGroupName	New name for the group. Only include this if changing the group's name. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	No
NewPath	New path for the group. Only include this if changing the group's path. Type: String Length constraints: Minimum length of 1. Maximum length of 512.	No

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
EntityAlreadyExists	The request was rejected because it attempted to create a resource that already exists.	409
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=UpdateGroup
&GroupName=Test
&NewGroupName=Test_1
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<UpdateGroupResponse>
   <UpdateGroupResult>
      <Group xmlns="http://iam.amazonaws.com/doc/2010-05-08/">
         <Path>/division_abc/subdivision_xyz/product_1234/engineering/</Path>
         <GroupName>Test_1</GroupName>
         <GroupId>AGP2MAB8DPLSRHEXAMPLE</GroupId>
         <Arn>arn:aws:iam::123456789012:group/division_abc/subdivision_xyz/
         product_1234/engineering/Test_1</Arn>
      </Group>
   </UpdateGroupResult>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</UpdateGroupResponse>
```

UpdateLoginProfile

Description

Changes the password for the specified user.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
Password	The new password for the user name. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes
UserName	Name of the user whose password you want to update. Type: String Length constraints: Minimum length of 1. Maximum length of 64.	Yes

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
EntityTemporarilyUnmodifiable	The request was rejected because it referenced an entity that is temporarily unmodifiable, such as a user name that was deleted and then recreated. The error indicates that the request is likely to succeed if you try again after waiting several minutes. The error message describes the entity.	409
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404
PasswordPolicyViolation	The request was rejected because the provided password did not meet the requirements imposed by the account password policy.	400

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=UpdateLoginProfile
&UserName=Bob
```

```
&Password=NewPassword
&AUTHPARAMS
```

Sample Response

```
<UpdateLoginProfileResponse>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</UpdateLoginProfileResponse>
```

UpdateServerCertificate

Description

Updates the name and/or the path of the specified server certificate.

> ⚠️ **Important**
>
> You should understand the implications of changing a server certificate's 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.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
NewPath	The new path for the server certificate. Include this only if you are updating the server certificate's path. Type: String Length constraints: Minimum length of 1. Maximum length of 512.	No
NewServerCertificateName	The new name for the server certificate. Include this only if you are updating the server certificate's name. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	No
ServerCertificateName	The name of the server certificate that you want to update. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
EntityAlreadyExists	The request was rejected because it attempted to create a resource that already exists.	409
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=UpdateServerCertificate
&ServerCertificateName=ProdServerCert
&NewServerCertificateName=ProdServerCertName
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<UpdateServerCertificateResponse>
  <ResponseMetadata>
    <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
  </ResponseMetadata>
</UpdateServerCertificateResponse>
```

UpdateSigningCertificate

Description

Changes the status of the specified signing certificate from active to disabled, or vice versa. This action can be used to disable a user's signing certificate as part of a certificate rotation work flow.

If the `UserName` field is not specified, the UserName is determined implicitly based on the AWS Access Key ID used to sign the request. Because this action works for access keys under the AWS account, this API can be used to manage root credentials even if the AWS account has no associated users.

For information about rotating certificates, see Managing Keys and Certificates in *Using AWS Identity and Access Management*.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
CertificateId	The ID of the signing certificate you want to update. Type: String Length constraints: Minimum length of 24. Maximum length of 128.	Yes
Status	The status you want to assign to the certificate. `Active` means the certificate can be used for API calls to AWS, while `Inactive` means the certificate cannot be used. Type: String Valid Values: `Active` | `Inactive`	Yes
UserName	Name of the user the signing certificate belongs to. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	No

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
`NoSuchEntity`	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
```

```
?Action=UpdateSigningCertificate
&UserName=Bob
&CertificateId=TA7SMP42TDN5Z26OBPJE7EXAMPLE
&Status=Inactive
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<UpdateSigningCertificateResponse>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</UpdateSigningCertificateResponse>
```

UpdateUser

Description

Updates the name and/or the path of the specified user.

> ⚠️ **Important**
>
> You should 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.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
`NewPath`	New path for the user. Include this parameter only if you're changing the user's path. Type: String Length constraints: Minimum length of 1. Maximum length of 512.	No
`NewUserName`	New name for the user. Include this parameter only if you're changing the user's name. Type: String Length constraints: Minimum length of 1. Maximum length of 64.	No
`UserName`	Name of the user to update. If you're changing the name of the user, this is the original user name. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
`EntityAlreadyExists`	The request was rejected because it attempted to create a resource that already exists.	409

Error	Description	HTTP Status Code
EntityTemporarilyUnmodifiable	The request was rejected because it referenced an entity that is temporarily unmodifiable, such as a user name that was deleted and then recreated. The error indicates that the request is likely to succeed if you try again after waiting several minutes. The error message describes the entity.	409
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=UpdateUser
&UserName=Bob
&NewUserName=Robert
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<UpdateUserResponse>
   <UpdateUserResult>
   <User>
      <Path>/division_abc/subdivision_xyz/</Path>
      <UserName>Robert</UserName>
      <UserId>AIDACKCEVSQ6C2EXAMPLE</UserId>
      <Arn>arn:aws::123456789012:user/division_abc/subdivision_xyz/Robert
      </Arn>
   </User>
   </UpdateUserResult>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</UpdateUserResponse>
```

UploadServerCertificate

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 Limitations on IAM Entities in *Using AWS Identity and Access Management*.

Note

Because the body of the public key certificate, private key, and the certificate chain can be large, you should use POST rather than GET when calling `UploadServerCertificate`. For information about setting up signatures and authorization through the API, go to Signing AWS API Requests in the *AWS General Reference*. For general information about using the Query API with IAM, go to Making Query Requests in *Using IAM*.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
CertificateBody	The contents of the public key certificate in PEM-encoded format. Type: String Length constraints: Minimum length of 1. Maximum length of 16384.	Yes
CertificateChain	The contents of the certificate chain. This is typically a concatenation of the PEM-encoded public key certificates of the chain. Type: String Length constraints: Minimum length of 1. Maximum length of 2097152.	No
Path	The path for the server certificate. For more information about paths, see Identifiers for IAM Entities in *Using AWS Identity and Access Management*. This parameter is optional. If it is not included, it defaults to a slash (/). Type: String Length constraints: Minimum length of 1. Maximum length of 512.	No
PrivateKey	The contents of the private key in PEM-encoded format. Type: String Length constraints: Minimum length of 1. Maximum length of 16384.	Yes

Name	Description	Required
ServerCertificateName	The name for the server certificate. Do not include the path in this value. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	Yes

Response Elements

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

Name	Description
ServerCertificateMetadata	The meta information of the uploaded server certificate without its certificate body, certificate chain, and private key. Type: ServerCertificateMetadata (p. 173)

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
EntityAlreadyExists	The request was rejected because it attempted to create a resource that already exists.	409
KeyPairMismatch	The request was rejected because the public key certificate and the private key do not match.	400
LimitExceeded	The request was rejected because it attempted to create resources beyond the current AWS account limits. The error message describes the limit exceeded.	409
MalformedCertificate	The request was rejected because the certificate was malformed or expired. The error message describes the specific error.	400

Examples

Sample Request

```
https://iam.amazonaws.com/
?Action=UploadServerCertificate
&ServerCertificateName=ProdServerCert
&Path=/company/servercerts/
&CertificateBody=-----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-----
&PrivateKey=-----BEGIN DSA PRIVATE KEY-----
MIIBugIBTTKBgQD33xToSXPJ6hr37L3+KNi3/7DgywlBcvlFPPSHIw3ORuO/22mT
8Cy5fT89WwNvZ3BPKWU6OZ38TQv3eWjNc/3U3+oqVNG2poX5nCPOtO1b96HYX2mR
3FTdH6FRKbQEhpDzZ6tRrjTHjMX6sT3JRWkBd2c4bGu+HUHO1H7QvrCTeQIVTKMs
TCKCyrLiGhUWuUGNJUMU6y6zToGTHl84Tz7TPwDGDXuy/Dk5s4jTVr+xibROC/gS
Qrs4Dzz3T1ze6lvU8S1KT9UsOB5FUJNTTPCPey+Lo4mmK6b23XdTyCIT8e2fsm2j
jHHC1pIPiTkdLS3j6ZYjF8LY6TENFng+LDY/xwPOl7TJVoD3J/WXC2J9CEYq9o34
kq6WWn3CgYTuo54nXUgnoCb3xdG8COFrg+oTbIkHTSzs3w5o/GGgKK7TDF3UlJjq
vHNyJQ6kWBrQRR1Xp5KYQ4c/Dm5kef+62mH53HpcCELguWVcffuVQpmq3EWL9Zp9
jobTJQ2VHjb5IVxiO6HRSd27di3njyrzUuJCyHSDTqwLJmTThpd6OTIUTL3Tc4m2
62TITdw53KWJEXAMPLE=
-----END DSA PRIVATE KEY-----
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```xml
<UploadServerCertificateResponse>
  <UploadServerCertificateResult>
    <ServerCertificateMetadata>
      <ServerCertificateName>ProdServerCert</ServerCertificateName>
      <Path>/company/servercerts/</Path>
     <Arn>arn:aws:iam::123456789012:server-certificate/company/servercerts/Prod
ServerCert</Arn>
      <UploadDate>2010-05-08T01:02:03.004Z</UploadDate>
      <ServerCertificateId>ASCACKCEVSQ6C2EXAMPLE</ServerCertificateId>
    </ServerCertificateMetadata>
  </UploadServerCertificateResult>
  <ResponseMetadata>
    <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
  </ResponseMetadata>
</UploadServerCertificateResponse>
```

UploadSigningCertificate

Description

Uploads an X.509 signing certificate and associates it with the specified user. Some AWS services use X.509 signing certificates to validate requests that are signed with a corresponding private key. When you upload the certificate, its default status is `Active`.

If the `UserName` field is not specified, the user name is determined implicitly based on the AWS Access Key ID used to sign the request. Because this action works for access keys under the AWS account, this API can be used to manage root credentials even if the AWS account has no associated users.

> 📕 **Note**
>
> Because the body of a X.509 certificate can be large, you should use POST rather than GET when calling `UploadSigningCertificate`. For information about setting up signatures and authorization through the API, go to Signing AWS API Requests in the *AWS General Reference*. For general information about using the Query API with IAM, go to Making Query Requests in *Using IAM*.

Request Parameters

For information about the common parameters that all actions use, see Common Query Parameters (p. 177).

Name	Description	Required
CertificateBody	The contents of the signing certificate. Type: String Length constraints: Minimum length of 1. Maximum length of 16384.	Yes
UserName	Name of the user the signing certificate is for. Type: String Length constraints: Minimum length of 1. Maximum length of 128.	No

Response Elements

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

Name	Description
Certificate	Information about the certificate. Type: SigningCertificate (p. 174)

Errors

For information about the common errors that all actions use, see Common Errors (p. 179).

Error	Description	HTTP Status Code
DuplicateCertificate	The request was rejected because the same certificate is associated to another user under the account.	409
EntityAlreadyExists	The request was rejected because it attempted to create a resource that already exists.	409
InvalidCertificate	The request was rejected because the certificate is invalid.	400
LimitExceeded	The request was rejected because it attempted to create resources beyond the current AWS account limits. The error message describes the limit exceeded.	409
MalformedCertificate	The request was rejected because the certificate was malformed or expired. The error message describes the specific error.	400
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404

Examples

Sample Request

```
POST / HTTP/1.1
Host: iam.amazonaws.com
Content-Type: application/x-www-form-urlencoded

Action=UploadSigningCertificate
&UserName=Bob
&CertificateBody=-----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-----
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response

```
<UploadSigningCertificateResponse>
   <UploadSigningCertificateResult>
      <Certificate>
         <UserName>Bob</UserName>
         <CertificateId>TA7SMP42TDN5Z26OBPJE7EXAMPLE</CertificateId>
         <CertificateBody>-----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-----</CertificateBody>
         <Status>Active</Status>
      </Certificate>
   </UploadSigningCertificateResult>
   <ResponseMetadata>
      <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
   </ResponseMetadata>
</UploadSigningCertificateResponse>
```

Data Types

The AWS Identity and Access Management API contains several data types that various actions use. This section describes each data type in detail.

![note icon] **Note**

> The order of each element in the response is not guaranteed. Applications should not assume a particular order.

The following data types are supported:

AccessKey

Description

The AccessKey data type contains information about an AWS access key.

This data type is used as a response element in the actions CreateAccessKey (p. 11) and ListAccessKeys (p. 80).

Contents

Name	Description
AccessKeyId	The ID for this access key. Type: String Length constraints: Minimum length of 16. Maximum length of 32.
CreateDate	The date when the access key was created. Type: DateTime
SecretAccessKey	The secret key used to sign requests. Type: String
Status	The status of the access key. `Active` means the key is valid for API calls, while `Inactive` means it is not. Type: String Valid Values: `Active` | `Inactive`
UserName	Name of the user the key is associated with. Type: String Length constraints: Minimum length of 1. Maximum length of 64.

AccessKeyMetadata

Description

The AccessKey data type contains information about an AWS access key, without its secret key.

This data type is used as a response element in the action ListAccessKeys (p. 80).

Contents

Name	Description
AccessKeyId	The ID for this access key. Type: String Length constraints: Minimum length of 16. Maximum length of 32.
CreateDate	The date when the access key was created. Type: DateTime
Status	The status of the access key. `Active` means the key is valid for API calls, while `Inactive` means it is not. Type: String Valid Values: `Active` | `Inactive`
UserName	Name of the user the key is associated with. Type: String Length constraints: Minimum length of 1. Maximum length of 64.

CreateAccessKeyResult

Description

Contains the result of a successful invocation of the CreateAccessKey (p. 11) action.

Contents

Name	Description
AccessKey	Information about the access key. Type: AccessKey (p. 151)

CreateGroupResult

Description

Contains the result of a successful invocation of the CreateGroup (p. 14) action.

Contents

Name	Description
Group	Information about the group. Type: Group (p. 160)

CreateInstanceProfileResult

Description

Contains the result of a successful invocation of the CreateInstanceProfile (p. 16) action.

Contents

Name	Description
InstanceProfile	Information about the instance profile. Type: InstanceProfile (p. 161)

CreateLoginProfileResult

Description

Contains the result of a successful invocation of the CreateLoginProfile (p. 18) action.

Contents

Name	Description
LoginProfile	The user name and password create date. Type: LoginProfile (p. 169)

CreateRoleResult

Description

Contains the result of a successful invocation of the CreateRole (p. 20) action.

Contents

Name	Description
Role	Information about the role. Type: Role (p. 171)

CreateUserResult

Description

Contains the result of a successful invocation of the CreateUser (p. 22) action.

Contents

Name	Description
User	Information about the user. Type: User (p. 175)

CreateVirtualMFADeviceResult

Description

Contains the result of a successful invocation of the CreateVirtualMFADevice (p. 24) action.

Contents

Name	Description
VirtualMFADevice	A newly created virtual MFA device. Type: VirtualMFADevice (p. 176)

GetAccountPasswordPolicyResult

Description

Contains the result of a successful invocation of the GetAccountPasswordPolicy (p. 56) action.

Contents

Name	Description
PasswordPolicy	The PasswordPolicy data type contains information about the account password policy. This data type is used as a response element in the action GetAccountPasswordPolicy (p. 56). Type: PasswordPolicy (p. 170)

GetAccountSummaryResult

Description

Contains the result of a successful invocation of the GetAccountSummary (p. 58) action.

Contents

Name	Description
`SummaryMap`	A set of key value pairs containing account-level information. `SummaryMap` contains the following keys: • `AccessKeysPerUserQuota` - Maximum number of access keys that can be created per user • `AccountMFAEnabled` - 1 if the root account has an MFA device assigned to it, 0 otherwise • `AssumeRolePolicySizeQuota` - Maximum allowed size for assume role policy documents (in kilobytes) • `GroupPolicySizeQuota` - Maximum allowed size for Group policy documents (in kilobytes) • `Groups` - Number of Groups for the AWS account • `GroupsPerUserQuota` - Maximum number of groups a user can belong to • `GroupsQuota` - Maximum groups allowed for the AWS account • `InstanceProfiles` - Number of instance profiles for the AWS account • `InstanceProfilesQuota` - Maximum instance profiles allowed for the AWS account • `MFADevices` - Number of MFA devices, either assigned or unassigned • `MFADevicesInUse` - Number of MFA devices that have been assigned to an IAM user or to the root account • `RolePolicySizeQuota` - Maximum allowed size for role policy documents (in kilobytes) • `Roles` - Number of roles for the AWS account • `RolesQuota` - Maximum roles allowed for the AWS account • `ServerCertificates` - Number of server certificates for the AWS account • `ServerCertificatesQuota` - Maximum server certificates allowed for the AWS account • `SigningCertificatesPerUserQuota` - Maximum number of X509 certificates allowed for a user • `UserPolicySizeQuota` - Maximum allowed size for user policy documents (in kilobytes) • `Users` - Number of users for the AWS account • `UsersQuota` - Maximum users allowed for the AWS account Type: String to Integer map

GetGroupPolicyResult

Description

Contains the result of a successful invocation of the GetGroupPolicy action.

Contents

Name	Description
GroupName	The group the policy is associated with. Type: String Length constraints: Minimum length of 1. Maximum length of 128.
PolicyDocument	The policy document. Type: String Length constraints: Minimum length of 1. Maximum length of 131072.
PolicyName	The name of the policy. Type: String Length constraints: Minimum length of 1. Maximum length of 128.

GetGroupResult

Description

Contains the result of a successful invocation of the GetGroup (p. 61) action.

Contents

Name	Description
Group	Information about the group. Type: Group (p. 160)
IsTruncated	A flag that indicates whether there are more user names to list. If your results were truncated, you can make a subsequent pagination request using the Marker request parameter to retrieve more user names in the list. Type: Boolean
Marker	If IsTruncated is true, then this element is present and contains the value to use for the Marker parameter in a subsequent pagination request. Type: String Length constraints: Minimum length of 1. Maximum length of 320.
Users	A list of users in the group. Type: User (p. 175) list

GetInstanceProfileResult

Description

Contains the result of a successful invocation of the GetInstanceProfile (p. 66) action.

Contents

Name	Description
InstanceProfile	Information about the instance profile. Type: InstanceProfile (p. 161)

GetLoginProfileResult

Description

Contains the result of a successful invocation of the GetLoginProfile (p. 68) action.

Contents

Name	Description
LoginProfile	User name and password create date for the user. Type: LoginProfile (p. 169)

GetRolePolicyResult

Description

Contains the result of a successful invocation of the GetRolePolicy (p. 72) action.

Contents

Name	Description
PolicyDocument	The policy document. Type: String Length constraints: Minimum length of 1. Maximum length of 131072.
PolicyName	The name of the policy. Type: String Length constraints: Minimum length of 1. Maximum length of 128.
RoleName	The role the policy is associated with. Type: String Length constraints: Minimum length of 1. Maximum length of 64.

GetRoleResult

Description

Contains the result of a successful invocation of the GetRole (p. 70) action.

Contents

Name	Description
Role	Information about the role. Type: Role (p. 171)

GetServerCertificateResult

Description

Contains the result of a successful invocation of the GetServerCertificate (p. 74) action.

Contents

Name	Description
ServerCertificate	Information about the server certificate. Type: ServerCertificate (p. 172)

GetUserPolicyResult

Description

Contains the result of a successful invocation of the GetUserPolicy (p. 78) action.

Contents

Name	Description
PolicyDocument	The policy document. Type: String Length constraints: Minimum length of 1. Maximum length of 131072.
PolicyName	The name of the policy. Type: String Length constraints: Minimum length of 1. Maximum length of 128.

Name	Description
UserName	The user the policy is associated with. Type: String Length constraints: Minimum length of 1. Maximum length of 128.

GetUserResult

Description

Contains the result of a successful invocation of the GetUser (p. 76) action.

Contents

Name	Description
User	Information about the user. Type: User (p. 175)

Group

Description

The Group data type contains information about a group.

This data type is used as a response element in the following actions:

- CreateGroup (p. 14)
- GetGroup (p. 61)
- ListGroups (p. 87)

Contents

Name	Description
Arn	The Amazon Resource Name (ARN) specifying the group. For more information about ARNs and how to use them in policies, see Identifiers for IAM Entities in *Using AWS Identity and Access Management*. Type: String Length constraints: Minimum length of 20. Maximum length of 2048.
CreateDate	The date when the group was created. Type: DateTime

Name	Description
`GroupId`	The stable and unique string identifying the group. For more information about IDs, see Identifiers for IAM Entities in *Using AWS Identity and Access Management*. Type: String Length constraints: Minimum length of 16. Maximum length of 32.
`GroupName`	The name that identifies the group. Type: String Length constraints: Minimum length of 1. Maximum length of 128.
`Path`	Path to the group. For more information about paths, see Identifiers for IAM Entities in *Using AWS Identity and Access Management*. Type: String Length constraints: Minimum length of 1. Maximum length of 512.

InstanceProfile

Description

The InstanceProfile data type contains information about an instance profile.

This data type is used as a response element in the following actions:

- CreateInstanceProfile (p. 16)
- GetInstanceProfile (p. 66)
- ListInstanceProfiles (p. 91)
- ListInstanceProfilesForRole (p. 93)

Contents

Name	Description
`Arn`	The Amazon Resource Name (ARN) specifying the instance profile. For more information about ARNs and how to use them in policies, see Identifiers for IAM Entities in *Using AWS Identity and Access Management*. Type: String Length constraints: Minimum length of 20. Maximum length of 2048.
`CreateDate`	The date when the instance profile was created. Type: DateTime
`InstanceProfileId`	The stable and unique string identifying the instance profile. For more information about IDs, see Identifiers for IAM Entities in *Using AWS Identity and Access Management*. Type: String Length constraints: Minimum length of 16. Maximum length of 32.

Name	Description
InstanceProfileName	The name identifying the instance profile. Type: String Length constraints: Minimum length of 1. Maximum length of 128.
Path	Path to the instance profile. For more information about paths, see Identifiers for IAM Entities in *Using AWS Identity and Access Management*. Type: String Length constraints: Minimum length of 1. Maximum length of 512.
Roles	The role associated with the instance profile. Type: Role (p. 171) list

ListAccessKeysResult

Description

Contains the result of a successful invocation of the ListAccessKeys (p. 80) action.

Contents

Name	Description
AccessKeyMetadata	A list of access key metadata. Type: AccessKeyMetadata (p. 152) list
IsTruncated	A flag that indicates whether there are more keys to list. If your results were truncated, you can make a subsequent pagination request using the Marker request parameter to retrieve more keys in the list. Type: Boolean
Marker	If IsTruncated is true, this element is present and contains the value to use for the Marker parameter in a subsequent pagination request. Type: String Length constraints: Minimum length of 1. Maximum length of 320.

ListAccountAliasesResult

Description

Contains the result of a successful invocation of the ListAccountAliases (p. 83) action.

Contents

Name	Description
AccountAliases	A list of aliases associated with the account. Type: String list
IsTruncated	A flag that indicates whether there are more account aliases to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more account aliases in the list. Type: Boolean
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 Length constraints: Minimum length of 1. Maximum length of 320.

ListGroupPoliciesResult

Description

Contains the result of a successful invocation of the action.

Contents

Name	Description
IsTruncated	A flag that indicates whether there are more policy names to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more policy names in the list. Type: Boolean
Marker	If `IsTruncated` is `true`, this element is present and contains the value to use for the `Marker` parameter in a subsequent pagination request. Type: String Length constraints: Minimum length of 1. Maximum length of 320.
PolicyNames	A list of policy names. Type: String list

ListGroupsForUserResult

Description

Contains the result of a successful invocation of the action.

Contents

Name	Description
Groups	A list of groups. Type: Group (p. 160) list
IsTruncated	A flag that indicates whether there are more groups to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more groups in the list. Type: Boolean
Marker	If `IsTruncated` is `true`, this element is present and contains the value to use for the `Marker` parameter in a subsequent pagination request. Type: String Length constraints: Minimum length of 1. Maximum length of 320.

ListGroupsResult

Description

Contains the result of a successful invocation of the ListGroups (p. 87) action.

Contents

Name	Description
Groups	A list of groups. Type: Group (p. 160) list
IsTruncated	A flag that indicates whether there are more groups to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more groups in the list. Type: Boolean
Marker	If `IsTruncated` is `true`, this element is present and contains the value to use for the `Marker` parameter in a subsequent pagination request. Type: String Length constraints: Minimum length of 1. Maximum length of 320.

ListInstanceProfilesForRoleResult

Description

Contains the result of a successful invocation of the ListInstanceProfilesForRole (p. 93) action.

Contents

Name	Description
InstanceProfiles	A list of instance profiles. Type: InstanceProfile (p. 161) list
IsTruncated	A flag that indicates whether there are more instance profiles to list. If your results were truncated, you can make a subsequent pagination request using the Marker request parameter to retrieve more instance profiles in the list. Type: Boolean
Marker	If IsTruncated is true, this element is present and contains the value to use for the Marker parameter in a subsequent pagination request. Type: String Length constraints: Minimum length of 1. Maximum length of 320.

ListInstanceProfilesResult

Description

Contains the result of a successful invocation of the ListInstanceProfiles (p. 91) action.

Contents

Name	Description
InstanceProfiles	A list of instance profiles. Type: InstanceProfile (p. 161) list
IsTruncated	A flag that indicates whether there are more instance profiles to list. If your results were truncated, you can make a subsequent pagination request using the Marker request parameter to retrieve more instance profiles in the list. Type: Boolean
Marker	If IsTruncated is true, this element is present and contains the value to use for the Marker parameter in a subsequent pagination request. Type: String Length constraints: Minimum length of 1. Maximum length of 320.

ListMFADevicesResult

Description

Contains the result of a successful invocation of the ListMFADevices (p. 96) action.

Contents

Name	Description
IsTruncated	A flag that indicates whether there are more MFA devices to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more MFA devices in the list. Type: Boolean
MFADevices	A list of MFA devices. Type: MFADevice (p. 170) list
Marker	If `IsTruncated` is `true`, this element is present and contains the value to use for the `Marker` parameter in a subsequent pagination request. Type: String Length constraints: Minimum length of 1. Maximum length of 320.

ListRolePoliciesResult

Description

Contains the result of a successful invocation of the ListRolePolicies (p. 98) action.

Contents

Name	Description
IsTruncated	A flag that indicates whether there are more policy names to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more policy names in the list. Type: Boolean
Marker	If `IsTruncated` is `true`, this element is present and contains the value to use for the `Marker` parameter in a subsequent pagination request. Type: String Length constraints: Minimum length of 1. Maximum length of 320.
PolicyNames	A list of policy names. Type: String list

ListRolesResult

Description

Contains the result of a successful invocation of the ListRoles (p. 100) action.

Contents

Name	Description
IsTruncated	A flag that indicates whether there are more roles to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more roles in the list. Type: Boolean
Marker	If `IsTruncated` is `true`, this element is present and contains the value to use for the `Marker` parameter in a subsequent pagination request. Type: String Length constraints: Minimum length of 1. Maximum length of 320.
Roles	A list of roles. Type: Role (p. 171) list

ListServerCertificatesResult

Description

Contains the result of a successful invocation of the ListServerCertificates (p. 103) action.

Contents

Name	Description
IsTruncated	A flag that indicates whether there are more server certificates to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more server certificates in the list. Type: Boolean
Marker	If `IsTruncated` is `true`, this element is present and contains the value to use for the `Marker` parameter in a subsequent pagination request. Type: String Length constraints: Minimum length of 1. Maximum length of 320.
ServerCertificateMetadataList	A list of server certificates. Type: ServerCertificateMetadata (p. 173) list

ListSigningCertificatesResult

Description

Contains the result of a successful invocation of the ListSigningCertificates (p. 106) action.

Contents

Name	Description
`Certificates`	A list of the user's signing certificate information. Type: SigningCertificate (p. 174) list
`IsTruncated`	A flag that indicates whether there are more certificate IDs to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more certificates in the list. Type: Boolean
`Marker`	If `IsTruncated` is `true`, this element is present and contains the value to use for the `Marker` parameter in a subsequent pagination request. Type: String Length constraints: Minimum length of 1. Maximum length of 320.

ListUserPoliciesResult

Description

Contains the result of a successful invocation of the ListUserPolicies (p. 109) action.

Contents

Name	Description
`IsTruncated`	A flag that indicates whether there are more policy names to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more policy names in the list. Type: Boolean
`Marker`	If `IsTruncated` is `true`, this element is present and contains the value to use for the `Marker` parameter in a subsequent pagination request. Type: String Length constraints: Minimum length of 1. Maximum length of 320.
`PolicyNames`	A list of policy names. Type: String list

ListUsersResult

Description

Contains the result of a successful invocation of the ListUsers (p. 111) action.

Contents

Name	Description
IsTruncated	A flag that indicates whether there are more user names to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more users in the list. Type: Boolean
Marker	If `IsTruncated` is `true`, this element is present and contains the value to use for the `Marker` parameter in a subsequent pagination request. Type: String Length constraints: Minimum length of 1. Maximum length of 320.
Users	A list of users. Type: User (p. 175) list

ListVirtualMFADevicesResult

Description

Contains the result of a successful invocation of the ListVirtualMFADevices (p. 113) action.

Contents

Name	Description
IsTruncated	A flag that indicates whether there are more items to list. If your results were truncated, you can make a subsequent pagination request using the `Marker` request parameter to retrieve more items the list. Type: Boolean
Marker	If `IsTruncated` is `true`, this element is present and contains the value to use for the `Marker` parameter in a subsequent pagination request. Type: String Length constraints: Minimum length of 1. Maximum length of 320.
VirtualMFADevices	Type: VirtualMFADevice (p. 176) list

LoginProfile

Description

The LoginProfile data type contains the user name and password create date for a user.

This data type is used as a response element in the actions CreateLoginProfile (p. 18) and GetLoginProfile (p. 68).

Contents

Name	Description
CreateDate	The date when the password for the user was created. Type: DateTime
UserName	The name of the user, which can be used for signing into the AWS Management Console. Type: String Length constraints: Minimum length of 1. Maximum length of 64.

MFADevice

Description

The `MFADevice` data type contains information about an MFA device.

This data type is used as a response element in the action .

Contents

Name	Description
EnableDate	The date when the MFA device was enabled for the user. Type: DateTime
SerialNumber	The serial number that uniquely identifies the MFA device. For virtual MFA devices, the serial number is the device ARN. Type: String Length constraints: Minimum length of 9. Maximum length of 256.
UserName	The user with whom the MFA device is associated. Type: String Length constraints: Minimum length of 1. Maximum length of 64.

PasswordPolicy

Description

The PasswordPolicy data type contains information about the account password policy.

This data type is used as a response element in the action .

Contents

Name	Description
AllowUsersToChangePassword	Specifies whether to allow IAM users to change their own password. Type: Boolean
MinimumPasswordLength	Minimum length to require for IAM user passwords. Type: Integer
RequireLowercaseCharacters	Specifies whether to require lowercase characters for IAM user passwords. Type: Boolean
RequireNumbers	Specifies whether to require numbers for IAM user passwords. Type: Boolean
RequireSymbols	Specifies whether to require symbols for IAM user passwords. Type: Boolean
RequireUppercaseCharacters	Specifies whether to require uppercase characters for IAM user passwords. Type: Boolean

Role

Description

The Role data type contains information about a role.

This data type is used as a response element in the following actions:

Contents

Name	Description
Arn	The Amazon Resource Name (ARN) specifying the role. For more information about ARNs and how to use them in policies, see Identifiers for IAM Entities in *Using AWS Identity and Access Management*. Type: String Length constraints: Minimum length of 20. Maximum length of 2048.

Name	Description
AssumeRolePolicyDocument	The policy that grants an entity permission to assume the role. The returned policy is URL-encoded according to RFC 3986. For more information about RFC 3986, go to http://www.faqs.org/rfcs/rfc3986.html. Type: String Length constraints: Minimum length of 1. Maximum length of 131072.
CreateDate	The date when the role was created. Type: DateTime
Path	Path to the role. For more information about paths, see Identifiers for IAM Entities in *Using AWS Identity and Access Management*. Type: String Length constraints: Minimum length of 1. Maximum length of 512.
RoleId	The stable and unique string identifying the role. For more information about IDs, see Identifiers for IAM Entities in *Using AWS Identity and Access Management*. Type: String Length constraints: Minimum length of 16. Maximum length of 32.
RoleName	The name identifying the role. Type: String Length constraints: Minimum length of 1. Maximum length of 64.

ServerCertificate

Description

The ServerCertificate data type contains information about a server certificate.

This data type is used as a response element in the action GetServerCertificate (p. 74).

Contents

Name	Description
CertificateBody	The contents of the public key certificate. Type: String Length constraints: Minimum length of 1. Maximum length of 16384.
CertificateChain	The contents of the public key certificate chain. Type: String Length constraints: Minimum length of 1. Maximum length of 2097152.

Name	Description
ServerCertificateMetadata	The meta information of the server certificate, such as its name, path, ID, and ARN. Type: ServerCertificateMetadata (p. 173)

ServerCertificateMetadata

Description

ServerCertificateMetadata contains information about a server certificate without its certificate body, certificate chain, and private key.

This data type is used as a response element in the action UploadServerCertificate (p. 144) and ListServerCertificates (p. 103).

Contents

Name	Description
Arn	The Amazon Resource Name (ARN) specifying the server certificate. For more information about ARNs and how to use them in policies, see Identifiers for IAM Entities in *Using AWS Identity and Access Management*. Type: String Length constraints: Minimum length of 20. Maximum length of 2048.
Path	Path to the server certificate. For more information about paths, see Identifiers for IAM Entities in *Using AWS Identity and Access Management*. Type: String Length constraints: Minimum length of 1. Maximum length of 512.
ServerCertificateId	The stable and unique string identifying the server certificate. For more information about IDs, see Identifiers for IAM Entities in *Using AWS Identity and Access Management*. Type: String Length constraints: Minimum length of 16. Maximum length of 32.
ServerCertificateName	The name that identifies the server certificate. Type: String Length constraints: Minimum length of 1. Maximum length of 128.
UploadDate	The date when the server certificate was uploaded. Type: DateTime

SigningCertificate

Description

The SigningCertificate data type contains information about an X.509 signing certificate.

This data type is used as a response element in the actions UploadSigningCertificate (p. 147) and ListSigningCertificates (p. 106).

Contents

Name	Description
CertificateBody	The contents of the signing certificate. Type: String Length constraints: Minimum length of 1. Maximum length of 16384.
CertificateId	The ID for the signing certificate. Type: String Length constraints: Minimum length of 24. Maximum length of 128.
Status	The status of the signing certificate. `Active` means the key is valid for API calls, while `Inactive` means it is not. Type: String Valid Values: `Active` | `Inactive`
UploadDate	The date when the signing certificate was uploaded. Type: DateTime
UserName	Name of the user the signing certificate is associated with. Type: String Length constraints: Minimum length of 1. Maximum length of 64.

UploadServerCertificateResult

Description

Contains the result of a successful invocation of the UploadServerCertificate (p. 144) action.

Contents

Name	Description
ServerCertificateMetadata	The meta information of the uploaded server certificate without its certificate body, certificate chain, and private key. Type: ServerCertificateMetadata (p. 173)

UploadSigningCertificateResult

Description

Contains the result of a successful invocation of the UploadSigningCertificate (p. 147) action.

Contents

Name	Description
Certificate	Information about the certificate. Type: SigningCertificate (p. 174)

User

Description

The User data type contains information about a user.

This data type is used as a response element in the following actions:

- CreateUser (p. 22)
- GetUser (p. 76)
- ListUsers (p. 111)

Contents

Name	Description
Arn	The Amazon Resource Name (ARN) specifying the user. For more information about ARNs and how to use them in policies, see Identifiers for IAM Entities in *Using AWS Identity and Access Management*. Type: String Length constraints: Minimum length of 20. Maximum length of 2048.
CreateDate	The date when the user was created. Type: DateTime
Path	Path to the user. For more information about paths, see Identifiers for IAM Entities in *Using AWS Identity and Access Management*. Type: String Length constraints: Minimum length of 1. Maximum length of 512.
UserId	The stable and unique string identifying the user. For more information about IDs, see Identifiers for IAM Entities in *Using AWS Identity and Access Management*. Type: String Length constraints: Minimum length of 16. Maximum length of 32.

Name	Description
UserName	The name identifying the user. Type: String Length constraints: Minimum length of 1. Maximum length of 64.

VirtualMFADevice

Description

The `VirtualMFADevice` data type contains information about a virtual MFA device.

Contents

Name	Description
Base32StringSeed	The Base32 seed defined as specified in RFC3548. The `Base32StringSeed` is Base64-encoded. Type: Blob
EnableDate	Type: DateTime
QRCodePNG	A QR code PNG image that encodes *otpauth://totp/$virtualMFADeviceName@$AccountName? secret=$Base32String* where $virtualMFADeviceName is one of the create call arguments, AccountName is the user name if set (accountId otherwise), and Base32String is the seed in Base32 format. The `Base32String` is Base64-encoded. Type: Blob
SerialNumber	The serial number associated with `VirtualMFADevice`. Type: String Length constraints: Minimum length of 9. Maximum length of 256.
User	The User data type contains information about a user. This data type is used as a response element in the following actions: • CreateUser (p. 22) • GetUser (p. 76) • ListUsers (p. 111) Type: User (p. 175)

Common Query Parameters

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

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

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

Common Errors

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

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

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