# Table of Contents

## Greengrass API

- CreateCoreDefinition ........................................................................................................ 8
- CreateCoreDefinitionVersion ............................................................................................... 12
- CreateDeployment ................................................................................................................ 15
- CreateDeviceDefinition ....................................................................................................... 18
- CreateDeviceDefinitionVersion ........................................................................................... 22
- CreateFunctionDefinition .................................................................................................... 25
- CreateFunctionDefinitionVersion ........................................................................................ 30
- CreateGroup ....................................................................................................................... 36
- CreateGroupCertificateAuthority ......................................................................................... 39
- CreateGroupVersion ............................................................................................................ 42
- CreateLoggerDefinition ........................................................................................................ 45
- CreateLoggerDefinitionVersion ........................................................................................... 49
- CreateResourceDefinition .................................................................................................... 52
- CreateResourceDefinitionVersion ....................................................................................... 58
- CreateSoftwareUpdateJob .................................................................................................... 64
- CreateSubscriptionDefinition ............................................................................................. 68
- CreateSubscriptionDefinitionVersion .................................................................................. 72
- DeleteCoreDefinition ......................................................................................................... 75
- DeleteDeviceDefinition ....................................................................................................... 77
- DeleteFunctionDefinition .................................................................................................... 79
- DeleteGroup ....................................................................................................................... 80
- DeleteLoggerDefinition ....................................................................................................... 82
- DeleteResourceDefinition ................................................................................................... 84
- DeleteSubscriptionDefinition ............................................................................................. 86
- DisassociateRoleFromGroup ................................................................................................. 87
- DisassociateServiceRoleFromAccount .................................................................................. 90
- GetAssociatedRole ............................................................................................................... 91
- GetConnectivityInfo ........................................................................................................... 94
- GetCoreDefinition .............................................................................................................. 97
- GetCoreDefinitionVersion ................................................................................................. 100
- GetDeploymentStatus .......................................................................................................... 103
- GetDeviceDefinition .......................................................................................................... 105
- GetDeviceDefinitionVersion ............................................................................................... 108
- GetFunctionDefinition ........................................................................................................ 111
- GetFunctionDefinitionVersion ........................................................................................... 113
- GetGroup ........................................................................................................................... 118
- GetGroupCertificateAuthority ............................................................................................. 120
- GetGroupCertificateConfiguration ......................................................................................... 123
- GetGroupVersion ............................................................................................................... 126
- GetLoggerDefinition .......................................................................................................... 129
- GetLoggerDefinitionVersion ............................................................................................... 132
- GetResourceDefinition ....................................................................................................... 135
- GetResourceDefinitionVersion ........................................................................................... 137
- GetServiceRoleForAccount ................................................................................................. 142
- GetSubscriptionDefinition ................................................................................................. 144
- GetSubscriptionDefinitionVersion ...................................................................................... 146
- ListCoreDefinitionVersions ............................................................................................... 149
- ListCoreDefinitions ............................................................................................................ 152
- ListDeployments ................................................................................................................. 154
- ListDeviceDefinitionVersions ............................................................................................ 157
<table>
<thead>
<tr>
<th>Endpoints</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>/greengrass/definition/cores</td>
<td>210</td>
</tr>
<tr>
<td>/greengrass/definition/cores/CoreDefinitionId</td>
<td>213</td>
</tr>
<tr>
<td>/greengrass/definition/cores/CoreDefinitionId/versions</td>
<td>219</td>
</tr>
<tr>
<td>/greengrass/definition/cores/CoreDefinitionId/versions/CoreDefinitionVersionId</td>
<td>225</td>
</tr>
<tr>
<td>/greengrass/definition/devices</td>
<td>232</td>
</tr>
<tr>
<td>/greengrass/definition/devices/DeviceDefinitionId</td>
<td>235</td>
</tr>
<tr>
<td>/greengrass/definition/devices/DeviceDefinitionId/versions</td>
<td>241</td>
</tr>
<tr>
<td>/greengrass/definition/devices/DeviceDefinitionId/versions/CoreDefinitionVersionId</td>
<td>247</td>
</tr>
<tr>
<td>/greengrass/definition/functions/FunctionDefinitionId</td>
<td>253</td>
</tr>
<tr>
<td>/greengrass/definition/functions/FunctionDefinitionId/versions</td>
<td>256</td>
</tr>
<tr>
<td>/greengrass/definition/functions/FunctionDefinitionId/versions/FunctionDefinitionVersionId</td>
<td>264</td>
</tr>
<tr>
<td>/greengrass/definition/loggers</td>
<td>270</td>
</tr>
<tr>
<td>/greengrass/definition/loggers/LoggerDefinitionId</td>
<td>278</td>
</tr>
<tr>
<td>/greengrass/definition/loggers/LoggerDefinitionId/versions</td>
<td>283</td>
</tr>
<tr>
<td>/greengrass/definition/loggers/LoggerDefinitionId/versions/CoreDefinitionVersionId</td>
<td>289</td>
</tr>
<tr>
<td>/greengrass/definition/loggers/LoggerDefinitionId/versions/LoggerDefinitionVersionId</td>
<td>295</td>
</tr>
<tr>
<td>/greengrass/definition/resources</td>
<td>302</td>
</tr>
<tr>
<td>/greengrass/definition/resources/ResourceDefinitionId</td>
<td>305</td>
</tr>
<tr>
<td>/greengrass/definition/resources/ResourceDefinitionId/versions</td>
<td>313</td>
</tr>
<tr>
<td>/greengrass/definition/resources/ResourceDefinitionId/versions/ResourceDefinitionVersionId</td>
<td>319</td>
</tr>
<tr>
<td>/greengrass/definition/subscriptions</td>
<td>328</td>
</tr>
<tr>
<td>/greengrass/definition/subscriptions/SubscriptionDefinitionId</td>
<td>333</td>
</tr>
<tr>
<td>/greengrass/definition/subscriptions/SubscriptionDefinitionId/versions</td>
<td>339</td>
</tr>
<tr>
<td>/greengrass/definition/subscriptions/SubscriptionDefinitionId/versions/SubscriptionDefinitionVersionId</td>
<td>345</td>
</tr>
<tr>
<td>SubscriptionDefinitionVersionId</td>
<td>352</td>
</tr>
<tr>
<td>/greengrass/groups</td>
<td>355</td>
</tr>
<tr>
<td>/greengrass/groups/GroupIId</td>
<td>361</td>
</tr>
<tr>
<td>/greengrass/groups/GroupIId/certificateauthorities</td>
<td>367</td>
</tr>
<tr>
<td>/greengrass/groups/GroupIId/certificateauthorities/configuration/expiry</td>
<td>373</td>
</tr>
<tr>
<td>/greengrass/groups/GroupIId/certificateauthorities/CertificateAuthorityIId</td>
<td>379</td>
</tr>
<tr>
<td>/greengrass/groups/GroupIId/deployments</td>
<td>382</td>
</tr>
<tr>
<td>/greengrass/groups/GroupIId/deployments/$reset</td>
<td>388</td>
</tr>
<tr>
<td>/greengrass/groups/GroupIId/deployments/DeploymentId/status</td>
<td>390</td>
</tr>
<tr>
<td>/greengrass/groups/GroupIId/role</td>
<td>393</td>
</tr>
</tbody>
</table>
/greengrass/groups/GroupId/versions ................................................................. 402
/greengrass/groups/GroupId/versions/GroupVersionId ...................................... 408
/greengrass/servicerole ....................................................................................... 411
/greengrass/things/ThingName/connectivityInfo ............................................. 417
/greengrass/updates ......................................................................................... 424

Parameters .......................................................................................................... 428
AssociateRoleToGroupRequestBody ............................................................... 429
AssociateServiceRoleToAccountRequestBody ................................................. 429
CertificateAuthorityId ...................................................................................... 430
CoreDefinitionId .............................................................................................. 430
CoreDefinitionVersionId .................................................................................. 430
CreateCoreDefinitionRequestBody ................................................................... 431
CreateCoreDefinitionVersionRequestBody ...................................................... 431
CreateDeploymentRequestBody ....................................................................... 432
CreateDeviceDefinitionRequestBody .............................................................. 433
CreateDeviceDefinitionVersionRequestBody .................................................. 433
CreateFunctionDefinitionRequestBody .......................................................... 434
CreateFunctionDefinitionVersionRequestBody ............................................. 435
CreateGroupRequestBody ................................................................................ 438
CreateGroupRequestBody .............................................................................. 438
CreateGroupVersionRequestBody .................................................................... 439
CreateLoggerDefinitionRequestBody .............................................................. 439
CreateLoggerDefinitionVersionRequestBody .................................................. 440
CreateResourceDefinitionRequestBody .......................................................... 441
CreateResourceDefinitionVersionRequestBody .............................................. 442
CreateSoftwareUpdateJobRequestBody .......................................................... 445
CreateSubscriptionDefinitionRequestBody ................................................... 446
CreateSubscriptionDefinitionVersionRequestBody ........................................ 447
DeploymentId ................................................................................................. 448
DeviceDefinitionId ......................................................................................... 448
DeviceDefinitionVersionId ............................................................................... 448
FunctionDefinitionId ...................................................................................... 449
FunctionDefinitionVersionId .......................................................................... 449
GroupId .......................................................................................................... 449
GroupVersionId ............................................................................................... 450
LoggerDefinitionId ......................................................................................... 450
LoggerDefinitionVersionId ............................................................................... 450
MaxResults ..................................................................................................... 450
NextToken ....................................................................................................... 451
ResetDeploymentsRequestBody ...................................................................... 451
ResourceDefinitionId .................................................................................... 452
ResourceDefinitionVersionId .......................................................................... 452
SubscriptionDefinitionId ................................................................................ 452
SubscriptionDefinitionVersionId ................................................................. 452
ThingName ..................................................................................................... 453
UpdateConnectivityInfoRequestBody ............................................................. 453
UpdateDefinitionRequestBody ...................................................................... 454
UpdateDeploymentRequestBody ...................................................................... 454
UpdateDeviceDefinitionRequestBody ............................................................ 454
UpdateGroupCertificateConfigurationRequestBody ...................................... 454
X-Amzn-Client-Token ..................................................................................... 455

Definitions ......................................................................................................... 455
AssociateRoleToGroupRequest ........................................................................ 457
AssociateServiceRoleToAccountRequest ....................................................... 457
AssociateServiceRoleToAccountResponse ..................................................... 458
ConnectivityInfo ............................................................................................ 458
Core ............................................................................................................... 458
CoreDefinitionVersion .................................................................................. 459
CreateDeploymentRequest .............................................................................. 460
CreateDeploymentResponse .......................................................................... 461
AWS Greengrass API

AWS Greengrass provides a resource-based API that uses Hypertext Application Language (HAL). HAL provides a standard way for expressing the resources and relationships of an API as hyperlinks. Using HAL, you use HTTP methods (GET, PUT, POST, DELETE) to submit requests and receive information about the API in the response. Applications can use the information returned to explore the functionality of the API. For more information about HAL, see the JSON Hypertext Application Language draft.

To request a HAL response from AWS Greengrass, specify application/hal+json for the accept request header.

Contents

- Actions (p. 1)
- Endpoints (p. 212)
- Parameters (p. 428)
- Definitions (p. 455)

Actions

The following actions are supported:

- AssociateRoleToGroup (p. 3)
- AssociateServiceRoleToAccount (p. 6)
- CreateCoreDefinition (p. 8)
- CreateCoreDefinitionVersion (p. 12)
- CreateDeployment (p. 15)
- CreateDeviceDefinition (p. 18)
- CreateDeviceDefinitionVersion (p. 22)
- CreateFunctionDefinition (p. 25)
- CreateFunctionDefinitionVersion (p. 30)
- CreateGroup (p. 36)
- CreateGroupCertificateAuthority (p. 39)
- CreateGroupVersion (p. 42)
- CreateLoggerDefinition (p. 45)
- CreateLoggerDefinitionVersion (p. 49)
- CreateResourceDefinition (p. 52)
- CreateResourceDefinitionVersion (p. 58)
- CreateSoftwareUpdateJob (p. 64)
- CreateSubscriptionDefinition (p. 68)
- CreateSubscriptionDefinitionVersion (p. 72)
- DeleteCoreDefinition (p. 75)
- DeleteDeviceDefinition (p. 77)
- DeleteFunctionDefinition (p. 79)
- DeleteGroup (p. 80)
- DeleteLoggerDefinition (p. 82)
- DeleteResourceDefinition (p. 84)
• DeleteSubscriptionDefinition (p. 86)
• DisassociateRoleFromGroup (p. 87)
• DisassociateServiceRoleFromAccount (p. 90)
• GetAssociatedRole (p. 91)
• GetConnectivityInfo (p. 94)
• GetCoreDefinition (p. 97)
• GetCoreDefinitionVersion (p. 100)
• GetDeploymentStatus (p. 103)
• GetDeviceDefinition (p. 105)
• GetDeviceDefinitionVersion (p. 108)
• GetFunctionDefinition (p. 111)
• GetFunctionDefinitionVersion (p. 113)
• GetGroup (p. 118)
• GetGroupCertificateAuthority (p. 120)
• GetGroupCertificateConfiguration (p. 123)
• GetGroupVersion (p. 126)
• GetLoggerDefinition (p. 129)
• GetLoggerDefinitionVersion (p. 132)
• GetResourceDefinition (p. 135)
• GetResourceDefinitionVersion (p. 137)
• GetServiceRoleForAccount (p. 142)
• GetSubscriptionDefinition (p. 144)
• GetSubscriptionDefinitionVersion (p. 146)
• ListCoreDefinitionVersions (p. 149)
• ListCoreDefinitions (p. 152)
• ListDeployments (p. 154)
• ListDeviceDefinitionVersions (p. 157)
• ListDeviceDefinitions (p. 160)
• ListFunctionDefinitionVersions (p. 162)
• ListFunctionDefinitions (p. 164)
• ListGroupCertificateAuthorities (p. 167)
• ListGroupVersions (p. 170)
• ListGroups (p. 172)
• ListLoggerDefinitionVersions (p. 174)
• ListLoggerDefinitions (p. 177)
• ListResourceDefinitionVersions (p. 179)
• ListResourceDefinitions (p. 182)
• ListSubscriptionDefinitionVersions (p. 184)
• ListSubscriptionDefinitions (p. 187)
• ResetDeployments (p. 189)
• UpdateConnectivityInfo (p. 191)
• UpdateCoreDefinition (p. 195)
• UpdateDeviceDefinition (p. 197)
• UpdateFunctionDefinition (p. 199)
• UpdateGroup (p. 201)
• UpdateGroupCertificateConfiguration (p. 203)
AssociateRoleToGroup

Associates a role with a group. Your AWS Greengrass core will use the role to access AWS cloud services. The role's permissions should allow Greengrass core Lambda functions to perform actions against the cloud.

URI: PUT /greengrass/groups/GroupId/role

CLI:

```
aws greengrass associate-role-to-group \
--group-id <value> \
[--role-arn <value>] \
[--cli-input-json <value>] \
[--generate-cli-skeleton]
```

cli-input-json format:

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

Parameters:

GroupId (p. 449)

The ID of the AWS Greengrass group.

where used: path; required: true

type: string

AssociateRoleToGroupRequestBody (p. 429)

where used: body; required: true

```
{
  "RoleArn": "string"
}
```

schema: AssociateRoleToGroupRequest

type: object

required: ["RoleArn"]

RoleArn

The ARN of the role you wish to associate with this group.

type: string
Responses:

200

success

AssociateRoleToGroupResponse (p. 494)

{
   "AssociatedAt": "string"
}

Group
type: object

AssociatedAt

The time, in milliseconds since the epoch, when the role ARN was associated with the group.
type: string

400

invalid request

GeneralError (p. 476)

{
   "Message": "string",
   "ErrorDetails": [
      {
         "DetailedErrorCode": "string",
         "DetailedErrorMessage": "string"
      }
   ]
}

GeneralError
type: object
required: ["Message"]

Message

A message containing information about the error.
type: string

ErrorDetails

A list of error details.
type: array
items: ErrorDetail (p. 468)

ErrorDetail
details about the error.
type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

500

server error

GeneralError (p. 476)

{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.
Associate a role with your account. AWS Greengrass will use the role to access your Lambda functions and AWS IoT resources. This is necessary for deployments to succeed. The role must have at least minimum permissions in the policy `AWSGreengrassResourceAccessRolePolicy`.

**URI:** `PUT /greengrass/servicerole`

**CLI:**
```
aws greengrass associate-service-role-to-account \
    [--role-arn <value>] \ 
    [--cli-input-json <value>] \ 
    [--generate-cli-skeleton]
```

**cli-input-json format:**
```
{
    "RoleArn": "string"
}
```

**Parameters:**

`AssociateServiceRoleToAccountRequestBody` (p. 429)

where used: body; required: true
```
{
    "RoleArn": "string"
}
```

**Responses:**

**200** success

`AssociateServiceRoleToAccountResponse` (p. 458)
```
{
```
"AssociatedAt": "string"
}

AssociateServiceRoleToAccountResponse

type: object

AssociatedAt

The time when the service role was associated with the account.

type: string

400

invalid request

GeneralError (p. 476)

{
   "Message": "string",
   "ErrorDetails": [
      {
         "DetailedErrorCode": "string",
         "DetailedErrorMessage": "string"
      }
   ]
}

GeneralError

   General error information.

type: object

   required: ["Message"]

Message

   A message containing information about the error.

type: string

ErrorDetails

   A list of error details.

type: array

   items: ErrorDetail (p. 468)

ErrorDetail

   Details about the error.

type: object

DetailedErrorCode

   A detailed error code.

type: string

DetailedErrorMessage

   A detailed error message.
CreateCoreDefinition

Create a core definition. You may provide the initial version of the core definition now or use CreateCoreDefinitionVersion at a later time. AWS Greengrass groups must each contain exactly one AWS Greengrass core.

URI: POST /greengrass/definition/cores
Produces: application/json

**CLI:**

```
aws greengrass create-core-definition \
  [--name <value>] \
  [--initial-version <value>] \
  [--amzn-client-token <value>] \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```
{
  "Name": "string",
  "InitialVersion": {
    "Cores": [
      {
        "Id": "string",
        "ThingArn": "string",
        "CertificateArn": "string",
        "SyncShadow": "boolean"
      }
    ],
    "AmznClientToken": "string"
  }
}
```

**Parameters:**

*CreateCoreDefinitionRequestBody* (p. 431)

Information needed to create a core definition.

where used: body; required: true

```
{
  "Name": "string",
  "InitialVersion": {
    "Cores": [
      {
        "Id": "string",
        "ThingArn": "string",
        "CertificateArn": "string",
        "SyncShadow": true
      }
    ]
  }
}
```

schema:

**Name**

The name of the core definition.

type: string

**InitialVersion**

Information about a core definition version.
CreateCoreDefinition

type: object

Cores

A list of cores in the core definition version.

type: array

items: Core (p. 458)

Core

Information about a core.

required: ["Id", "ThingArn", "CertificateArn"]

Id

The ID of the core.

type: string

ThingArn

The ARN of the thing which is the core.

type: string

CertificateArn

The ARN of the certificate associated with the core.

type: string

SyncShadow

If true, the core's local shadow is automatically synced with the cloud.

type: boolean

X-Amzn-Client-Token (p. 455)

A client token used to correlate requests and responses.

where used: header; required: false

type: string

Responses:

200 (CreateCoreDefinitionResponse)

DefinitionInformation (p. 463)

```json
{
    "Name": "string",
    "Id": "string",
    "Arn": "string",
    "LastUpdatedTimestamp": "string",
    "CreationTimestamp": "string",
    "LatestVersion": "string",
    "LatestVersionArn": "string"
}
```
DefinitionInformation

Information about a definition.

type: object

Name

The name of the definition.

type: string

Id

The ID of the definition.

type: string

Arn

The ARN of the definition.

type: string

LastUpdatedTimestamp

The time, in milliseconds since the epoch, when the definition was last updated.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the definition was created.

type: string

LatestVersion

The latest version of the definition.

type: string

LatestVersionArn

The ARN of the latest version of the definition.

type: string

400

invalid request

GeneralError (p. 476)

{
   "Message": "string",
   "ErrorDetails": [
      {
         "DetailedErrorCode": "string",
         "DetailedErrorMessage": "string"
      }
   ]
}

GeneralError

General error information.
CreateCoreDefinitionVersion

Creates a version of a core definition that has already been defined. AWS Greengrass groups must each contain exactly one AWS Greengrass core.

URI: POST /greengrass/definition/cores/CoreDefinitionId/versions

Produces: application/json

CLI:

```
aws greengrass create-core-definition-version
   --core-definition-id <value> \ 
   [--cores <value>] \ 
   [--amzn-client-token <value>] \ 
   [--cli-input-json <value>] \ 
   [--generate-cli-skeleton]
```

cli-input-json format:

```
{
   "CoreDefinitionId": "string",
   "Cores": [
       {
           "Id": "string",
           "ThingArn": "string",
           "CertificateArn": "string",
```
Parameters:

CoreDefinitionId (p. 430)

The ID of the core definition.

where used: path; required: true
type: string

CreateCoreDefinitionVersionRequestBody (p. 431)

where used: body; required: true

```json
{
    "Cores": [
    {
        "Id": "string",
        "ThingArn": "string",
        "CertificateArn": "string",
        "SyncShadow": true
    }
    ]
}
```

schema:
CoreDefinitionVersion

Information about a core definition version.
type: object

Cores

A list of cores in the core definition version.
type: array
items: Core (p. 458)

Core

Information about a core.
type: object

required: ["Id", "ThingArn", "CertificateArn"]

Id

The ID of the core.
type: string

ThingArn

The ARN of the thing which is the core.
type: string
CertificateArn

The ARN of the certificate associated with the core.

  type: string
SyncShadow

If true, the core's local shadow is automatically synced with the cloud.

  type: boolean

X-Amzn-Client-Token (p. 455)

A client token used to correlate requests and responses.

  where used: header; required: false
  type: string

Responses:

200 (CreateCoreDefinitionVersionResponse)

VersionInformation (p. 521)

```
{
  "Arn": "string",
  "Id": "string",
  "Version": "string",
  "CreationTimestamp": "string"
}
```

VersionInformation

Information about a version.

  type: object
Arn

The ARN of the version.

  type: string
Id

The ID of the version.

  type: string
Version

The unique ID of the version.

  type: string
CreationTimestamp

The time, in milliseconds since the epoch, when the version was created.

  type: string

400

invalid request
GeneralError (p. 476)

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

**GeneralError**

General error information.

- **type**: object

  - **required**: ["Message"]

**Message**

A message containing information about the error.

- **type**: string

**ErrorDetails**

A list of error details.

- **type**: array

  - **items**: ErrorDetail (p. 468)

**ErrorDetail**

Details about the error.

- **type**: object

  - **DetailedErrorCode**

    A detailed error code.

    - **type**: string

  - **DetailedErrorMessage**

    A detailed error message.

    - **type**: string

---

**CreateDeployment**

Create a deployment.

- **URI**: POST /greengrass/groups/GroupId/deployments

- **Produces**: application/json

**CLI:**

```
aws greengrass create-deployment \
```
CreateDeployment

--group-id <value> \
[--deployment-type <value>] \
[--deployment-id <value>] \
[--group-version-id <value>] \
[--amzn-client-token <value>] \
[--cli-input-json <value>] \
[--generate-cli-skeleton]

cli-input-json format:

{
  "GroupId": "string",
  "DeploymentType": "NewDeployment|Redeployment|ResetDeployment|ForceResetDeployment",
  "DeploymentId": "string",
  "GroupVersionId": "string",
  "AmznClientToken": "string"
}

Parameters:

GroupId (p. 449)

The ID of the AWS Greengrass group.

where used: path; required: true

type: string

CreateDeploymentRequestBody (p. 432)

where used: body; required: true

{
  "DeploymentType": "NewDeployment|Redeployment|ResetDeployment|ForceResetDeployment",
  "DeploymentId": "string",
  "GroupVersionId": "string"
}

schema:
CreateDeploymentRequest

   Information about a deployment.

type: object

   required: ["DeploymentType"]

DeploymentType

type: string

   enum: ["NewDeployment", "Redeployment", "ResetDeployment", "ForceResetDeployment"]

DeploymentId

   The ID of the deployment if you wish to redeploy a previous deployment.

type: string

GroupVersionId

   The ID of the group version to be deployed.
**CreateDeployment**

**X-Amzn-Client-Token (p. 455)**

A client token used to correlate requests and responses.

where used: header; required: false

type: string

**Responses:**

**200**

Success. The group was deployed.

CreateDeploymentResponse (p. 461)

```json
{
  "DeploymentArn": "string",
  "DeploymentId": "string"
}
```

CreateDeploymentResponse

type: object

DeploymentArn

The ARN of the deployment.

type: string

DeploymentId

The ID of the deployment.

type: string

**400**

invalid request

GeneralError (p. 476)

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

GeneralError

General error information.

type: object

required: ["Message"]
CreateDeviceDefinition

Creates a device definition. You may provide the initial version of the device definition now or use CreateDeviceDefinitionVersion at a later time.

URI: POST /greengrass/definition/devices

Produces: application/json

CLI:

```bash
aws greengrass create-device-definition \
[--name <value>] \ 
[--initial-version <value>] \ 
[--amzn-client-token <value>] \ 
[--cli-input-json <value>] \ 
[--generate-cli-skeleton]
```

cli-input-json format:

```json
{
    "Name": "string",
    "InitialVersion": {
        "Devices": [
            {
                "Id": "string",
                "ThingArn": "string",
                "CertificateArn": "string",
                "SyncShadow": "boolean"
            }
        ]
    }
}
```
CreateDeviceDefinition

Parameters:

CreateDeviceDefinitionRequestBody (p. 433)

Information needed to create a device definition.

where used: body; required: true

```json
{
   "Name": "string",
   "InitialVersion": {
      "Devices": [
         {
            "Id": "string",
            "ThingArn": "string",
            "CertificateArn": "string",
            "SyncShadow": true
         }
      ]
   }
}
```

schema:

Name

The name of the device definition.

type: string

InitialVersion

Information about a device definition version.

type: object

Devices

A list of devices in the definition version.

type: array

items: Device (p. 465)

Device

Information about a device.

type: object

required: ["Id", "ThingArn", "CertificateArn"]

Id

The ID of the device.

type: string

ThingArn

The thing ARN of the device.
CertificateArn

The ARN of the certificate associated with the device.

SyncShadow

If true, the device's local shadow will be automatically synced with the cloud.

X-Amzn-Client-Token (p. 455)

A client token used to correlate requests and responses.

Responses:

200 (CreateDeviceDefinitionResponse)

DefinitionInformation (p. 463)

```
{
   "Name": "string",
   "Id": "string",
   "Arn": "string",
   "LastUpdatedTimestamp": "string",
   "CreationTimestamp": "string",
   "LatestVersion": "string",
   "LatestVersionArn": "string"
}
```

DefinitionInformation

Information about a definition.

type: object

Name

The name of the definition.

type: string

Id

The ID of the definition.

type: string

Arn

The ARN of the definition.

type: string

LastUpdatedTimestamp

The time, in milliseconds since the epoch, when the definition was last updated.
type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the definition was created.
type: string

LatestVersion

The latest version of the definition.
type: string

LatestVersionArn

The ARN of the latest version of the definition.
type: string

400

invalid request

**GeneralError (p. 476)**

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

**GeneralError**

General error information.
type: object

required: ["Message"]

**Message**

A message containing information about the error.
type: string

**ErrorDetails**

A list of error details.
type: array

items:** ErrorDetail (p. 468)**

**ErrorDetail**

Details about the error.
type: object

**DetailedErrorCode**

A detailed error code.
CreateDeviceDefinitionVersion

Creates a version of a device definition that has already been defined.

URI: POST /greengrass/definition/devices/{DeviceDefinitionId}/versions

Produces: application/json

CLI:

```
aws greengrass create-device-definition-version \
  --device-definition-id <value> \
  [--devices <value>] \
  [--amzn-client-token <value>] \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```
{
  "DeviceDefinitionId": "string",
  "Devices": [
    {
      "Id": "string",
      "ThingArn": "string",
      "CertificateArn": "string",
      "SyncShadow": "boolean"
    }
  ],
  "AmznClientToken": "string"
}
```

Parameters:

**DeviceDefinitionId** (p. 448)

The ID of the device definition.

where used: path; required: true

type: string

**CreateDeviceDefinitionVersionRequestBody** (p. 433)

where used: body; required: true

```
{
  "Devices": [
    {
      "Id": "string",
      "ThingArn": "string",
      "CertificateArn": "string",
      "SyncShadow": "boolean"
    }
  ]
}
```
"SyncShadow": true
]
"

schema:
DeviceDefinitionVersion
   Information about a device definition version.
   type: object
Devices
   A list of devices in the definition version.
   type: array
   items: Device (p. 465)
Device
   Information about a device.
   type: object
   required: ["Id", "ThingArn", "CertificateArn"]
Id
   The ID of the device.
   type: string
ThingArn
   The thing ARN of the device.
   type: string
CertificateArn
   The ARN of the certificate associated with the device.
   type: string
SyncShadow
   If true, the device's local shadow will be automatically synced with the cloud.
   type: boolean
X-Amzn-Client-Token (p. 455)
   A client token used to correlate requests and responses.
   where used: header; required: false
   type: string

Responses:

200 (CreateDeviceDefinitionVersionResponse)
   VersionInformation (p. 521)
CreateDeviceDefinitionVersion

```json
{
   "Arn": "string",
   "Id": "string",
   "Version": "string",
   "CreationTimestamp": "string"
}
```

**VersionInformation**

Information about a version.

type: object

**Arn**

The ARN of the version.

type: string

**Id**

The ID of the version.

type: string

**Version**

The unique ID of the version.

type: string

**CreationTimestamp**

The time, in milliseconds since the epoch, when the version was created.

type: string

**400**

invalid request

**GeneralError** (p. 476)

```json
{
   "Message": "string",
   "ErrorDetails": [
      {
         "DetailedErrorCode": "string",
         "DetailedErrorMessage": "string"
      }
   ]
}
```

**GeneralError**

General error information.

type: object

required: ["Message"]

**Message**

A message containing information about the error.
CreateFunctionDefinition

Creates a Lambda function definition which contains a list of Lambda functions and their configurations to be used in a group. You can create an initial version of the definition by providing a list of Lambda functions and their configurations now, or use CreateFunctionDefinitionVersion later.

URI: POST /greengrass/definition/functions

Produces: application/json

CLI:

```bash
aws greengrass create-function-definition \
[--name <value>] \
[--initial-version <value>] \
[--amzn-client-token <value>] \
[--cli-input-json <value>] \
[--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "Name": "string",
  "InitialVersion": {
    "Functions": [
      {
        "Id": "string",
        "FunctionArn": "string",
        "FunctionConfiguration": {
          "Pinned": "boolean",
          "Executable": "string",
          "ExecArgs": "string",
          "MemorySize": "integer",
          "Timeout": "integer"
        }
      }
    ]
  }
}
```
Parameters:

createFunctionDefinitionRequestBody (p. 434)

where used: body; required: true

```json
{
  "Name": "string",
  "InitialVersion": {
    "Functions": [
      {
        "Id": "string",
        "FunctionArn": "string",
        "FunctionConfiguration": {
          "Pinned": true,
          "Executable": "string",
          "ExecArgs": "string",
          "MemorySize": 0,
          "Timeout": 0,
          "EncodingType": "binary|json",
          "Environment": {
            "Variables": {
              "additionalProperty0": "string",
              "additionalProperty1": "string",
              "additionalProperty2": "string"
            },
            "ResourceAccessPolicies": [
              {
                "ResourceId": "string",
                "Permission": "ro|rw"
              }
            ],
            "AccessSysfs": true
          }
        }
      }
    ]
  }
}
```

schema:
Name
   The name of the function definition.
   type: string

InitialVersion
   Information about a function definition version.
   type: object

Functions
   A list of Lambda functions in this function definition version.
   type: array
   items: Function (p. 469)

   Information about a Lambda function.
   type: object
   required: ["Id"]

Id
   The ID of the Lambda function.
   type: string

FunctionArn
   The ARN of the Lambda function.
   type: string

FunctionConfiguration
   The configuration of the Lambda function.
   type: object

Pinned
   True if the function is pinned. Pinned means the function is long-lived and starts when the core starts.
   type: boolean

Executable
   The name of the function executable.
   type: string

ExecArgs
   The execution arguments.
   type: string

MemorySize
   The memory size, in KB, which the function requires.
   type: integer
Timeout

The allowed function execution time, after which Lambda should terminate the function. This timeout still applies to pinned lambdas for each request.

type: integer

EncodingType

The expected encoding type of the input payload for the function. The default is json.

type: string
enum: ["binary", "json"]

Environment

The environment configuration of the function.

type: object

Variables

Environment variables for the Lambda function's configuration.

type: object
additionalProperties: "type"=>"string"

ResourceAccessPolicies

A list of the resources, with their permissions, to which the Lambda function will be granted access. A Lambda function can have at most 10 resources.

type: array
items: ResourceAccessPolicy (p. 511)

ResourceAccessPolicy

A policy used by the function to access a resource.

type: object
required: ["ResourceId"]

ResourceId

The ID of the resource. (This ID is assigned to the resource when you create the resource definition.)

type: string

Permission

The type of permission a function has to access a resource.

type: string
enum: ["ro", "rw"]

AccessSysfs

If true, the Lambda function is allowed to access the host's /sys folder. Use this when the Lambda function needs to read device information from /sys.

type: boolean
**X-Amzn-Client-Token (p. 455)**

A client token used to correlate requests and responses.
where used: header; required: false
type: string

**Responses:**

200 (CreateFunctionDefinitionResponse)

**DefinitionInformation (p. 463)**

```json
{
 "Name": "string",
 "Id": "string",
 "Arn": "string",
 "LastUpdatedTimestamp": "string",
 "CreationTimestamp": "string",
 "LatestVersion": "string",
 "LatestVersionArn": "string"
}
```

DefinitionInformation

Information about a definition.
type: object

Name

The name of the definition.
type: string

Id

The ID of the definition.
type: string

Arn

The ARN of the definition.
type: string

LastUpdatedTimestamp

The time, in milliseconds since the epoch, when the definition was last updated.
type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the definition was created.
type: string

LatestVersion

The latest version of the definition.
type: string
LatestVersionArn

The ARN of the latest version of the definition.
type: string

400
invalid request

GeneralError (p. 476)

```
{
  "Message": "string",
  "ErrorDetails": [ 
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

GeneralError

General error information.
type: object
required: ["Message"]
Message

A message containing information about the error.
type: string
ErrorDetails

A list of error details.
type: array
items: ErrorDetail (p. 468)
ErrorDetail

Details about the error.
type: object
DetailedErrorCode

A detailed error code.
type: string
DetailedErrorMessage

A detailed error message.
type: string

CreateFunctionDefinitionVersion

Creates a version of a Lambda function definition that has already been defined.
URL: POST /greengrass/definition/functions/{FunctionDefinitionId}/versions

Produces: application/json

CLI:

```sh/aws greengrass create-function-definition-version \
--function-definition-id <value> \
[--functions <value>] \
[--amzn-client-token <value>] \
[--cli-input-json <value>] \
[--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "FunctionDefinitionId": "string",
  "Functions": [
    {
      "Id": "string",
      "FunctionArn": "string",
      "FunctionConfiguration": {
        "Pinned": "boolean",
        "Executable": "string",
        "ExecArgs": "string",
        "MemorySize": "integer",
        "Timeout": "integer",
        "EncodingType": "binary|json",
        "Environment": {
          "Variables": {
            "additionalProperty0": "string",
            "additionalProperty1": "string",
            "additionalProperty2": "string"
          },
          "ResourceAccessPolicies": [
            {
              "ResourceId": "string",
              "Permission": "ro|rw"
            }
          ],
          "AccessSysfs": "boolean"
        }
      }
    }
  ],
  "AmznClientToken": "string"
}
```

Parameters:

**FunctionDefinitionId (p. 449)**

The ID of the Lambda function definition.

where used: path; required: true

type: string

CreateFunctionDefinitionVersionRequestBody (p. 435)

Information about the function definition version.
where used: body; required: true

```
{
   "Functions": [
       {
         "Id": "string",
         "FunctionArn": "string",
         "FunctionConfiguration": {
             "Pinned": true,
             "Executable": "string",
             "ExecArgs": "string",
             "MemorySize": 0,
             "Timeout": 0,
             "EncodingType": "binary|json",
             "Environment": {
                 "Variables": {
                     "additionalProperty0": "string",
                     "additionalProperty1": "string",
                     "additionalProperty2": "string"
                 },
                 "ResourceAccessPolicies": [
                     {
                         "ResourceId": "string",
                         "Permission": "ro|rw"
                     }
                 ],
                 "AccessSysfs": true
             }
         }
       }
   ]
}
```

schema:
FunctionDefinitionVersion

Information about a function definition version.

- **type**: object

Functions

A list of Lambda functions in this function definition version.

- **type**: array

  - **items**: Function (p. 469)

  Information about a Lambda function.

  - **type**: object

  - **required**: ["Id"]

    **Id**

    The ID of the Lambda function.

    - **type**: string

    FunctionArn

    The ARN of the Lambda function.

    - **type**: string
FunctionConfiguration
   The configuration of the Lambda function.
   type: object

Pinned
   True if the function is pinned. Pinned means the function is long-lived and starts when the core
   starts.
   type: boolean

Executable
   The name of the function executable.
   type: string

ExecArgs
   The execution arguments.
   type: string

MemorySize
   The memory size, in KB, which the function requires.
   type: integer

Timeout
   The allowed function execution time, after which Lambda should terminate the function. This
   timeout still applies to pinned lambdas for each request.
   type: integer

EncodingType
   The expected encoding type of the input payload for the function. The default is json.
   type: string
   enum: ["binary", "json"]

Environment
   The environment configuration of the function.
   type: object

Variables
   Environment variables for the Lambda function's configuration.
   type: object
   additionalProperties: "%type"=>"string"

ResourceAccessPolicies
   A list of the resources, with their permissions, to which the Lambda function will be granted
   access. A Lambda function can have at most 10 resources.
   type: array
   items: ResourceAccessPolicy (p. 511)
ResourceAccessPolicy

A policy used by the function to access a resource.

type: object

required: ["ResourceId"]

ResourceId

The ID of the resource. (This ID is assigned to the resource when you create the resource definition.)

type: string

Permission

The type of permission a function has to access a resource.

type: string

enum: ["ro", "rw"]

AccessSysfs

If true, the Lambda function is allowed to access the host's /sys folder. Use this when the Lambda function needs to read device information from /sys.

type: boolean

X-Amzn-Client-Token (p. 455)

A client token used to correlate requests and responses.

where used: header; required: false

type: string

Responses:

200 (CreateFunctionDefinitionVersionResponse)

VersionInformation (p. 521)

```
{
  "Arn": "string",
  "Id": "string",
  "Version": "string",
  "CreationTimestamp": "string"
}
```

VersionInformation

Information about a version.

type: object

Arn

The ARN of the version.

type: string
Id
  The ID of the version.
  type: string

Version
  The unique ID of the version.
  type: string

CreationTimestamp
  The time, in milliseconds since the epoch, when the version was created.
  type: string

400
  invalid request

GeneralError (p. 476)

  {
    "Message": "string",
    "ErrorDetails": [
      {
        "DetailedErrorCode": "string",
        "DetailedErrorMessage": "string"
      }
    ]
  }

GeneralError
  General error information.
  type: object
  required: ["Message"]

Message
  A message containing information about the error.
  type: string

ErrorDetails
  A list of error details.
  type: array
  items: ErrorDetail (p. 468)

ErrorDetail
  Details about the error.
  type: object

DetailedErrorCode
  A detailed error code.
  type: string
DetailedErrorMessage
A detailed error message.

type: string

CreateGroup

Creates a group. You may provide the initial version of the group or use CreateGroupVersion at a later time.

**Tip**
You can use the `gg_group_setup` package (from GitHub) as a library or command-line application to create and deploy AWS Greengrass groups.

**URI:** POST /greengrass/groups

**Produces:** application/json

**CLI:**

```bash
aws greengrass create-group \
  [--name <value>] \
  [--initial-version <value>] \
  [--amzn-client-token <value>] \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "Name": "string",
  "InitialVersion": {
    "CoreDefinitionVersionArn": "string",
    "DeviceDefinitionVersionArn": "string",
    "FunctionDefinitionVersionArn": "string",
    "SubscriptionDefinitionVersionArn": "string",
    "LoggerDefinitionVersionArn": "string",
    "ResourceDefinitionVersionArn": "string"
  },
  "AmznClientToken": "string"
}
```

**Parameters:**

`CreateGroupRequestBody` (p. 438)

where used: body; required: true

```json
{
  "Name": "string",
  "InitialVersion": {
    "CoreDefinitionVersionArn": "string",
    "DeviceDefinitionVersionArn": "string",
    "FunctionDefinitionVersionArn": "string",
    "SubscriptionDefinitionVersionArn": "string",
    "LoggerDefinitionVersionArn": "string",
    "ResourceDefinitionVersionArn": "string"
  }
}
```
CreateGroup

schema:
Name
   The name of the group.
   type: string
InitialVersion
   Information about a group version.
   type: object
CoreDefinitionVersionArn
   The ARN of the core definition version for this group.
   type: string
DeviceDefinitionVersionArn
   The ARN of the device definition version for this group.
   type: string
FunctionDefinitionVersionArn
   The ARN of the function definition version for this group.
   type: string
SubscriptionDefinitionVersionArn
   The ARN of the subscription definition version for this group.
   type: string
LoggerDefinitionVersionArn
   The ARN of the logger definition version for this group.
   type: string
ResourceDefinitionVersionArn
   The resource definition version ARN for this group.
   type: string
X-Amzn-Client-Token (p. 455)
   A client token used to correlate requests and responses.
   where used: header; required: false
   type: string

Responses:

200 (CreateGroupResponse)
   Success. The group was created.
   DefinitionInformation (p. 463)
CreateGroup

```json
{
    "Name": "string",
    "Id": "string",
    "Arn": "string",
    "LastUpdatedTimestamp": "string",
    "CreationTimestamp": "string",
    "LatestVersion": "string",
    "LatestVersionArn": "string"
}
```

**DefinitionInformation**

Information about a definition.

- **Name**
  - type: string
  - The name of the definition.

- **Id**
  - type: string
  - The ID of the definition.

- **Arn**
  - type: string
  - The ARN of the definition.

- **LastUpdatedTimestamp**
  - type: string
  - The time, in milliseconds since the epoch, when the definition was last updated.

- **CreationTimestamp**
  - type: string
  - The time, in milliseconds since the epoch, when the definition was created.

- **LatestVersion**
  - type: string
  - The latest version of the definition.

- **LatestVersionArn**
  - type: string
  - The ARN of the latest version of the definition.

**400**

Invalid request

**GeneralError (p. 476)**

```json
{
    "Message": "string",
    "ErrorDetails": [
        {
            ...  
        }
    ]
}
```
AWS Greengrass API Reference
CreateGroupCertificateAuthority

"DetailedErrorCode": "string",
"DetailedErrorMessage": "string"
]
}
}

GeneralError
General error information.
type: object
required: ["Message"]

Message
A message containing information about the error.
type: string

ErrorDetails
A list of error details.
type: array
items: ErrorDetail (p. 468)

ErrorDetail
Details about the error.
type: object

DetailedErrorCode
A detailed error code.
type: string

DetailedErrorMessage
A detailed error message.
type: string

CreateGroupCertificateAuthority
Creates a CA for the group. If a CA already exists, it will rotate the existing CA.

URI: POST /greengrass/groups/GroupId/certificateauthorities

Produces: application/json

CLI:

aws greengrass create-group-certificate-authority \
--group-id <value> \
[--amzn-client-token <value>] \
[--cli-input-json <value>] \
[--generate-cli-skeleton]

cli-input-json format:
Parameters:

**GroupId (p. 449)**

The ID of the AWS Greengrass group.

where used: path; required: true

type: string

**X-Amzn-Client-Token (p. 455)**

A client token used to correlate requests and responses.

where used: header; required: false

type: string

Responses:

**200**

Success. The response body contains the new active CA ARN.

CreateGroupCertificateAuthorityResponse (p. 461)

```json
{
  "GroupCertificateAuthorityArn": "string"
}
```

CreateGroupCertificateAuthorityResponse

- type: object
- GroupCertificateAuthorityArn
  - The ARN of the group certificate authority.
  - type: string

**400**

invalid request

GeneralError (p. 476)

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```
GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

500

server error

GeneralError (p. 476)

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.
CreateGroupVersion

Creates a version of a group which has already been defined.

URI: POST /greengrass/groups/GroupId/versions

Produces: application/json

CLI:

```bash
aws greengrass create-group-version
  --group-id <value>
  [ --core-definition-version-arn <value> ]
  [ --device-definition-version-arn <value> ]
  [ --function-definition-version-arn <value> ]
  [ --subscription-definition-version-arn <value> ]
  [ --logger-definition-version-arn <value> ]
  [ --resource-definition-version-arn <value> ]
  [ --amzn-client-token <value> ]
  [ --cli-input-json <value> ]
  [ --generate-cli-skeleton ]
```

cli-input-json format:

```json
{
  "GroupId": "string",
  "CoreDefinitionVersionArn": "string",
  "DeviceDefinitionVersionArn": "string",
  "FunctionDefinitionVersionArn": "string",
  "SubscriptionDefinitionVersionArn": "string",
  "LoggerDefinitionVersionArn": "string",
  "ResourceDefinitionVersionArn": "string",
  "AmznClientToken": "string"
}
```
Parameters:

**GroupId (p. 449)**

The ID of the AWS Greengrass group.

where used: path; required: true

type: string

**CreateGroupVersionRequestBody (p. 438)**

where used: body; required: true

```json
{
    "CoreDefinitionVersionArn": "string",
    "DeviceDefinitionVersionArn": "string",
    "FunctionDefinitionVersionArn": "string",
    "SubscriptionDefinitionVersionArn": "string",
    "LoggerDefinitionVersionArn": "string",
    "ResourceDefinitionVersionArn": "string"
}
```

schema:

GroupVersion

Information about a group version.

type: object

CoreDefinitionVersionArn

The ARN of the core definition version for this group.

type: string

DeviceDefinitionVersionArn

The ARN of the device definition version for this group.

type: string

FunctionDefinitionVersionArn

The ARN of the function definition version for this group.

type: string

SubscriptionDefinitionVersionArn

The ARN of the subscription definition version for this group.

type: string

LoggerDefinitionVersionArn

The ARN of the logger definition version for this group.

type: string

ResourceDefinitionVersionArn

The resource definition version ARN for this group.

type: string
X-Amzn-Client-Token (p. 455)

A client token used to correlate requests and responses.

where used: header; required: false
type: string

Responses:

200 (CreateGroupVersionResponse)

Success. The response contains information about the group version.

VersionInformation (p. 521)

```
{
  "Arn": "string",
  "Id": "string",
  "Version": "string",
  "CreationTimestamp": "string"
}
```

VersionInformation

Information about a version.
type: object

 Arn

The ARN of the version.
type: string

 Id

The ID of the version.
type: string

 Version

The unique ID of the version.
type: string

 CreationTimestamp

The time, in milliseconds since the epoch, when the version was created.
type: string

400

invalid request

GeneralError (p. 476)

```
{
  "Message": "string",
  "ErrorDetails": [
    {
      "Message": "string",
      "Code": "string",
      "Region": "string",
      "RequestId": "string",
      "ErrorDetail": "string"
    }
  ]
}
```


CreateLoggerDefinition

Creates a logger definition. You may provide the initial version of the logger definition now or use CreateLoggerDefinitionVersion at a later time.

URL: POST /greengrass/definition/loggers

Produces: application/json

CLI:

```bash
aws greengrass create-logger-definition \
  [--name <value>] \
  [--initial-version <value>] \
  [--amzn-client-token <value>] \
  [--cli-input-json <value>] \
```
cli-input-json format:

```json
{
    "Name": "string",
    "InitialVersion": {
        "Loggers": [ 
            { "Id": "string",
              "Type": "FileSystem|AWSCloudWatch",
              "Component": "GreengrassSystem|Lambda",
              "Level": "DEBUG|INFO|WARN|ERROR|FATAL",
              "Space": "integer"
            } 
        ],
        "AmznClientToken": "string"
    }
}
```

Parameters:

CreateLoggerDefinitionRequestBody (p. 439)

where used: body; required: true

```json
{
    "Name": "string",
    "InitialVersion": {
        "Loggers": [ 
            { "Id": "string",
              "Type": "FileSystem|AWSCloudWatch",
              "Component": "GreengrassSystem|Lambda",
              "Level": "DEBUG|INFO|WARN|ERROR|FATAL",
              "Space": 0
            } 
        ]
    }
}
```

schema:

Name

The name of the logger definition.

type: string

InitialVersion

Information about a logger definition version.

type: object

Loggers

A list of loggers.

type: array

items: Logger (p. 505)
Logger

Information about a logger
type: object
required: ["Id", "Type", "Component", "Level"]

Id
The id of the logger.
type: string

Type
type: string
enum: ["FileSystem", "AWSCloudWatch"]

Component
type: string
enum: ["GreengrassSystem", "Lambda"]

Level
type: string
enum: ["DEBUG", "INFO", "WARN", "ERROR", "FATAL"]

Space
The amount of file space, in KB, to use if the local file system is used for logging purposes.
type: integer

X-Amzn-Client-Token (p. 455)
A client token used to correlate requests and responses.
where used: header; required: false
type: string

Responses:

200 (CreateLoggerDefinitionResponse)

DefinitionInformation (p. 463)

{
  "Name": "string",
  "Id": "string",
  "Arn": "string",
  "LastUpdatedTimestamp": "string",
  "CreationTimestamp": "string",
  "LatestVersion": "string",
  "LatestVersionArn": "string"
}

DefinitionInformation
Information about a definition.
type: object
Name
The name of the definition.
type: string
Id
The ID of the definition.
type: string
Arn
The ARN of the definition.
type: string
LastUpdatedTimestamp
The time, in milliseconds since the epoch, when the definition was last updated.
type: string
CreationTimestamp
The time, in milliseconds since the epoch, when the definition was created.
type: string
LatestVersion
The latest version of the definition.
type: string
LatestVersionArn
The ARN of the latest version of the definition.
type: string

400
invalid request
GeneralError (p. 476)

{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}

GeneralError
General error information.
type: object
required: ["Message"]
CreateLoggerDefinitionVersion

Creates a version of a logger definition that has already been defined.

URI: POST /greengrass/definition/loggers/\{LoggerDefinitionId\}/versions

Produces: application/json

CLI:

```
aws greengrass create-logger-definition-version
  --logger-definition-id <value> \n  [--loggers <value>] \n  [--amzn-client-token <value>] \n  [--cli-input-json <value>] \n  [--generate-cli-skeleton]
```

cli-input-json format:

```
{
  "LoggerDefinitionId": "string",
  "Loggers": [
    {
      "Id": "string",
      "Type": "FileSystem|AWSCloudWatch",
      "Component": "GreengrassSystem|Lambda",
      "Level": "DEBUG|INFO|WARN|ERROR|FATAL",
      "Space": "integer"
    }
  ],
}
```
"AmznClientToken": "string"
}

**Parameters:**

**LoggerDefinitionId (p. 450)**

The ID of the logger definition.

where used: path; required: true
type: string

**CreateLoggerDefinitionVersionRequestBody (p. 440)**

where used: body; required: true

```json
{
  "Loggers": [
    {
      "Id": "string",
      "Type": "FileSystem|AWSCloudWatch",
      "Component": "GreengrassSystem|Lambda",
      "Level": "DEBUG/INFO/WARN/ERROR/FATAL",
      "Space": 0
    }
  ]
}
```

**schema:**

**LoggerDefinitionVersion**

Information about a logger definition version.

type: object

**Loggers**

A list of loggers.

type: array

items: **Logger (p. 505)**

**Logger**

Information about a logger

type: object

required: ["Id", "Type", "Component", "Level"]

**Id**

The id of the logger.

type: string

**Type**

type: string
enum: ["FileSystem", "AWSCloudWatch"]
Component
type: string
enum: ["GreengrassSystem", "Lambda"]

Level
type: string
enum: ["DEBUG", "INFO", "WARN", "ERROR", "FATAL"]

Space
The amount of file space, in KB, to use if the local file system is used for logging purposes.

type: integer

X-Amzn-Client-Token (p. 455)
A client token used to correlate requests and responses.

where used: header; required: false
type: string

Responses:

200 (CreateLoggerDefinitionVersionResponse)

VersionInformation (p. 521)

```json
{
  "Arn": "string",
  "Id": "string",
  "Version": "string",
  "CreationTimestamp": "string"
}
```

VersionInformation
Information about a version.
type: object

Arn
The ARN of the version.
type: string

Id
The ID of the version.
type: string

Version
The unique ID of the version.
type: string

CreationTimestamp
The time, in milliseconds since the epoch, when the version was created.
CreateResourceDefinition

Create a resource definition which contains a list of resources to be used in a group. You can create an initial version of the definition by providing a list of resources now, or use CreateResourceDefinitionVersion later.
CreateResourceDefinition

URI: POST /greengrass/definition/resources

Produces: application/json

CLI:

```
aws greengrass create-resource-definition \
    [--name <value>] \ 
    [--initial-version <value>] \ 
    [--amzn-client-token <value>] \ 
    [--cli-input-json <value>] \ 
    [--generate-cli-skeleton]
```

cli-input-json format:

```
{
    "Name": "string",
    "InitialVersion": {
        "Resources": [
            {
                "Id": "string",
                "Name": "string",
                "ResourceDataContainer": {
                    "LocalDeviceResourceData": {
                        "SourcePath": "string",
                        "GroupOwnerSetting": {
                            "AutoAddGroupOwner": "boolean",
                            "GroupOwner": "string"
                        }
                    },
                    "LocalVolumeResourceData": {
                        "SourcePath": "string",
                        "DestinationPath": "string",
                        "GroupOwnerSetting": {
                            "AutoAddGroupOwner": "boolean",
                            "GroupOwner": "string"
                        }
                    },
                    "SageMakerMachineLearningModelResourceData": {
                        "SageMakerJobArn": "string",
                        "DestinationPath": "string"
                    },
                    "S3MachineLearningModelResourceData": {
                        "S3Uri": "string",
                        "DestinationPath": "string"
                    }
                }
            }
        ],
        "AmznClientToken": "string"
    }
}
```

Parameters:

CreateResourceDefinitionRequestBody (p. 441)

where used: body; required: true
"Name": "string",
"InitialVersion": {
    "Resources": [
        {
            "Id": "string",
            "Name": "string",
            "ResourceDataContainer": {
                "--- NOTE:": "choose one of the following ---",
                "LocalDeviceResourceData": {
                    "SourcePath": "string",
                    "GroupOwnerSetting": {
                        "AutoAddGroupOwner": true,
                        "GroupOwner": "string"
                    }
                },
                "LocalVolumeResourceData": {
                    "SourcePath": "string",
                    "DestinationPath": "string",
                    "GroupOwnerSetting": {
                        "AutoAddGroupOwner": true,
                        "GroupOwner": "string"
                    }
                },
                "SageMakerMachineLearningModelResourceData": {
                    "SageMakerJobArn": "string",
                    "DestinationPath": "string"
                },
                "S3MachineLearningModelResourceData": {
                    "S3Uri": "string",
                    "DestinationPath": "string"
                }
            }
        }
    ]
}
}

schema:
Name

The name of the resource definition.

type: string

InitialVersion

Information about a resource definition version.

type: object

Resources

A list of resources.

type: array

items: Resource (p. 508)

Resource

Information about a resource.

type: object

required: ["Id", "Name", "ResourceDataContainer"]
Id

The resource ID, used to refer to a resource in the Lambda function configuration. Max length is 128 characters with pattern [a-zA-Z0-9_:.-]+. This must be unique within a Greengrass group.

type: string

Name

The descriptive resource name, which is displayed on the AWS Greengrass console. Max length 128 characters with pattern [a-zA-Z0-9_:.-]+. This must be unique within a Greengrass group.

type: string

ResourceDataContainer

A container for resource data. The container takes only one of the following supported resource data types: LocalDeviceResourceData, LocalVolumeResourceData, SageMakerMachineLearningModelResourceData, S3MachineLearningModelResourceData.

type: object

NOTE: oneOf

LocalDeviceResourceData

Attributes that define a local device resource.

type: object

SourcePath

The local absolute path of the device resource. The source path for a device resource can refer only to a character device or block device under /dev.

type: string

GroupOwnerSetting

Group owner related settings for local resources.

type: object

AutoAddGroupOwner

If true, AWS Greengrass automatically adds the specified Linux OS group owner of the resource to the Lambda process privileges. Thus the Lambda process will have the file access permissions of the added Linux group.

type: boolean

GroupOwner

The name of the Linux OS group whose privileges will be added to the Lambda process. This field is optional.

type: string

LocalVolumeResourceData

Attributes that define a local volume resource.

type: object
SourcePath

The local absolute path of the volume resource on the host. The source path for a volume resource type cannot start with /proc or /sys.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.

type: string

GroupOwnerSetting

Group owner related settings for local resources.

type: object

AutoAddGroupOwner

If true, AWS Greengrass automatically adds the specified Linux OS group owner of the resource to the Lambda process privileges. Thus the Lambda process will have the file access permissions of the added Linux group.

type: boolean

GroupOwner

The name of the Linux OS group whose privileges will be added to the Lambda process. This field is optional.

type: string

SageMakerMachineLearningModelResourceData

Attributes that define an Amazon SageMaker machine learning resource.

type: object

SageMakerJobArn

The ARN of the Amazon SageMaker training job that represents the source model.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.

type: string

S3MachineLearningModelResourceData

Attributes that define an Amazon S3 machine learning resource.

type: object

S3Uri

The URI of the source model in an Amazon S3 bucket. The model package must be in tar.gz or .zip format.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.

type: string
**X-Amzn-Client-Token (p. 455)**

A client token used to correlate requests and responses.

where used: header; required: false

- **type**: string

**Responses:**

**200 (CreateResourceDefinitionResponse)**

**DefinitionInformation (p. 463)**

```json
{
"Name": "string",
"Id": "string",
"Arn": "string",
"LastUpdatedTimestamp": "string",
"CreationTimestamp": "string",
"LatestVersion": "string",
"LatestVersionArn": "string"
}
```

**DefinitionInformation**

Information about a definition.

- **type**: object

  **Name**
  
  The name of the definition.

  - **type**: string

  **Id**
  
  The ID of the definition.

  - **type**: string

  **Arn**
  
  The ARN of the definition.

  - **type**: string

  **LastUpdatedTimestamp**
  
  The time, in milliseconds since the epoch, when the definition was last updated.

  - **type**: string

  **CreationTimestamp**
  
  The time, in milliseconds since the epoch, when the definition was created.

  - **type**: string

  **LatestVersion**
  
  The latest version of the definition.

  - **type**: string
LatestVersionArn

The ARN of the latest version of the definition.

type: string

400

Invalid request

GeneralError (p. 476)

```
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

CreateResourceDefinitionVersion

Creates a version of a resource definition that has already been defined.
URI: POST /greengrass/definition/resources/<ResourceDefinitionId>/versions

Produces: application/json

CLI:

```
aws greengrass create-resource-definition-version \
  --resource-definition-id <value> \
  [--resources <value>] \
  [--amzn-client-token <value>] \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```
{
  "ResourceDefinitionId": "string",
  "Resources": [
    {
      "Id": "string",
      "Name": "string",
      "ResourceDataContainer": {
        "-- NOTE:": "choose one of the following --*",
        "LocalDeviceResourceData": {
          "SourcePath": "string",
          "GroupOwnerSetting": {
            "AutoAddGroupOwner": "boolean",
            "GroupOwner": "string"
          }
        },
        "LocalVolumeResourceData": {
          "SourcePath": "string",
          "DestinationPath": "string",
          "GroupOwnerSetting": {
            "AutoAddGroupOwner": "boolean",
            "GroupOwner": "string"
          }
        },
        "SageMakerMachineLearningModelResourceData": {
          "SageMakerJobArn": "string",
          "DestinationPath": "string"
        },
        "S3MachineLearningModelResourceData": {
          "S3Uri": "string",
          "DestinationPath": "string"
        }
      }
    }
  ],
  "AmznClientToken": "string"
}
```

Parameters:

**ResourceDefinitionId** (p. 452)

The ID of the resource definition.

where used: path; required: true

type: string
CreateResourceDefinitionVersionRequestBody (p. 442)

Information about the resource definition version.

where used: body; required: true

```json
{
  "Resources": [
    {
      "Id": "string",
      "Name": "string",
      "ResourceDataContainer": {
        "--- NOTE:": "choose one of the following --*
        "LocalDeviceResourceData": {
          "SourcePath": "string",
          "GroupOwnerSetting": {
            "AutoAddGroupOwner": true,
            "GroupOwner": "string"
          }
        },
        "LocalVolumeResourceData": {
          "SourcePath": "string",
          "DestinationPath": "string",
          "GroupOwnerSetting": {
            "AutoAddGroupOwner": true,
            "GroupOwner": "string"
          }
        },
        "SageMakerMachineLearningModelResourceData": {
          "SageMakerJobArn": "string",
          "DestinationPath": "string"
        },
        "S3MachineLearningModelResourceData": {
          "S3Uri": "string",
          "DestinationPath": "string"
        }
      }
    }
  ]
}
```

schema:

ResourceDefinitionVersion

Information about a resource definition version.

  type: object

Resources

A list of resources.

  type: array

  items: Resource (p. 508)

Resource

Information about a resource.

  type: object

  required: ["Id", "Name", "ResourceDataContainer"]
Id

The resource ID, used to refer to a resource in the Lambda function configuration. Max length is 128 characters with pattern [a-zA-Z0-9_:.-]+. This must be unique within a Greengrass group.

type: string
Name

The descriptive resource name, which is displayed on the AWS Greengrass console. Max length 128 characters with pattern [a-zA-Z0-9_:.-]+. This must be unique within a Greengrass group.

type: string
ResourceDataContainer

A container for resource data. The container takes only one of the following supported resource data types: LocalDeviceResourceData, LocalVolumeResourceData, SageMakerMachineLearningModelResourceData, S3MachineLearningModelResourceData.

type: object
NOTE: oneOf
LocalDeviceResourceData

Attributes that define a local device resource.

type: object
SourcePath

The local absolute path of the device resource. The source path for a device resource can refer only to a character device or block device under /dev.

type: string
GroupOwnerSetting

Group owner related settings for local resources.

type: object
AutoAddGroupOwner

If true, AWS Greengrass automatically adds the specified Linux OS group owner of the resource to the Lambda process privileges. Thus the Lambda process will have the file access permissions of the added Linux group.

type: boolean
GroupOwner

The name of the Linux OS group whose privileges will be added to the Lambda process. This field is optional.

type: string
LocalVolumeResourceData

Attributes that define a local volume resource.

type: object
SourcePath

The local absolute path of the volume resource on the host. The source path for a volume resource type cannot start with /proc or /sys.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.

type: string

GroupOwnerSetting

Group owner related settings for local resources.

type: object

AutoAddGroupOwner

If true, AWS Greengrass automatically adds the specified Linux OS group owner of the resource to the Lambda process privileges. Thus the Lambda process will have the file access permissions of the added Linux group.

type: boolean

GroupOwner

The name of the Linux OS group whose privileges will be added to the Lambda process. This field is optional.

type: string

SageMakerMachineLearningModelResourceData

Attributes that define an Amazon SageMaker machine learning resource.

type: object

SageMakerJobArn

The ARN of the Amazon SageMaker training job that represents the source model.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.

type: string

S3MachineLearningModelResourceData

Attributes that define an Amazon S3 machine learning resource.

type: object

S3Uri

The URI of the source model in an Amazon S3 bucket. The model package must be in tar.gz or .zip format.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.
X-Amzn-Client-Token (p. 455)

A client token used to correlate requests and responses.

where used: header; required: false

type: string

Responses:

200 (CreateResourceDefinitionVersionResponse)

VersionInformation (p. 521)

{
   "Arn": "string",
   "Id": "string",
   "Version": "string",
   "CreationTimestamp": "string"
}

VersionInformation

Information about a version.

type: object

Arn

The ARN of the version.

type: string

Id

The ID of the version.

type: string

Version

The unique ID of the version.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the version was created.

type: string

400

invalid request

GeneralError (p. 476)

{
   "Message": "string",
   "ErrorDetails": [
      {
      
```
CreateSoftwareUpdateJob

Creates a software update for a core or group of cores (specified as an IoT thing group.) Use this to update the OTA Agent as well as the Greengrass core software. It makes use of the IoT Jobs feature which provides additional commands to manage a Greengrass core software update job.

URI: POST /greengrass/updates

Produces: application/json

CLI:

```
aws greengrass create-software-update-job \[-update-targets-architecture <value>] \[-update-targets <value>] \[-software-to-update <value>]
```
CreateSoftwareUpdateJob

Parameters:

CreateSoftwareUpdateJobRequestBody (p. 445)

where used: body; required: true

```
{
  "UpdateTargetsArchitecture": "armv7l|x86_64|aarch64",
  "UpdateTargets": ["string"],
  "SoftwareToUpdate": "core|ota_agent",
  "S3UrlSignerRole": "string",
  "UpdateAgentLogLevel": "NONE|TRACE|DEBUG|VERBOSE|INFO|WARN|ERROR|FATAL",
  "UpdateTargetsOperatingSystem": "ubuntu|rpi_3|amazon_linux",
  "AmznClientToken": "string"
}
```

schema:

CreateSoftwareUpdateJobRequest

Request for the CreateSoftwareUpdateJob API.

type: object

required: ["UpdateTargetsArchitecture", "UpdateTargets", "SoftwareToUpdate",
  "S3UrlSignerRole", "UpdateTargetsOperatingSystem"]

UpdateTargetsArchitecture

The architecture of the cores which are the targets of an update.

type: string

enum: ["armv7l", "x86_64", "aarch64"]

UpdateTargets

The ARNs of the targets (IoT things or IoT thing groups) that this update will be applied to.

type: array
SoftwareToUpdate

The piece of software on the Greengrass core that will be updated.

type: string
enum: ["core", "ota_agent"]

S3UrlSignerRole

The IAM Role that Greengrass will use to create pre-signed URLs pointing towards the update artifact.

type: string

UpdateAgentLogLevel

The minimum level of log statements that should be logged by the OTA Agent during an update.

type: string
enum: ["NONE", "TRACE", "DEBUG", "VERBOSE", "INFO", "WARN", "ERROR", "FATAL"]

UpdateTargetsOperatingSystem

The operating system of the cores which are the targets of an update.

type: string
enum: ["ubuntu", "raspbian", "amazon_linux"]

X-Amzn-Client-Token (p. 455)

A client token used to correlate requests and responses.

where used: header; required: false
type: string

Responses:

200

success

CreateSoftwareUpdateJobResponse (p. 462)

```json
{
    "IotJobId": "string",
    "IotJobArn": "string"
}
```

CreateSoftwareUpdateJobResponse

type: object

IotJobId

The IoT Job Id corresponding to this update.

type: string
IoTJobArn

The IoT Job ARN corresponding to this update.

type: string

400

invalid request

GeneralError (p. 476)

```
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

500

server error

GeneralError (p. 476)
CreateSubscriptionDefinition

Creates a subscription definition. You may provide the initial version of the subscription definition now or use CreateSubscriptionDefinitionVersion at a later time.

**URI:** POST /greengrass/definition/subscriptions

**Produces:** application/json

**CLI:**

```
aws greengrass create-subscription-definition \
```
CreateSubscriptionDefinition

```
[--name <value>] \n[--initial-version <value>] \n[--amzn-client-token <value>] \n[--cli-input-json <value>] \n[--generate-cli-skeleton]
```

cli-input-json format:

```
{
  "Name": "string",
  "InitialVersion": {
    "Subscriptions": [
      {
        "Id": "string",
        "Source": "string",
        "Subject": "string",
        "Target": "string"
      }
    ],
    "AmznClientToken": "string"
  }
}
```

Parameters:

CreateSubscriptionDefinitionRequestBody (p. 446)

where used: body; required: true

```
{
  "Name": "string",
  "InitialVersion": {
    "Subscriptions": [
      {
        "Id": "string",
        "Source": "string",
        "Subject": "string",
        "Target": "string"
      }
    ]
  }
}
```

schema:

Name

The name of the subscription definition.

  type: string

InitialVersion

Information about a subscription definition version.

  type: object

Subscriptions

A list of subscriptions.

  type: array
items: Subscription (p. 518)

Subscription

Information about a subscription.

type: object

required: ["Id", "Source", "Subject", "Target"]

Id

The id of the subscription.

type: string

Source

The source of the subscription. Can be a thing ARN, a Lambda function ARN, 'cloud' (which represents the AWS IoT cloud), or 'GGShadowService'.

type: string

Subject

The subject of the message.

type: string

Target

Where the message is sent to. Can be a thing ARN, a Lambda function ARN, 'cloud' (which represents the AWS IoT cloud), or 'GGShadowService'.

type: string

X-Amzn-Client-Token (p. 455)

A client token used to correlate requests and responses.

where used: header; required: false

type: string

Responses:

200 (CreateSubscriptionDefinitionResponse)

DefinitionInformation (p. 463)

{
  "Name": "string",
  "Id": "string",
  "Arn": "string",
  "LastUpdatedTimestamp": "string",
  "CreationTimestamp": "string",
  "LatestVersion": "string",
  "LatestVersionArn": "string"
}
CreateSubscriptionDefinition

type: object

Name

The name of the definition.

type: string

Id

The ID of the definition.

type: string

Arn

The ARN of the definition.

type: string

LastUpdatedTimestamp

The time, in milliseconds since the epoch, when the definition was last updated.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the definition was created.

type: string

LatestVersion

The latest version of the definition.

type: string

LatestVersionArn

The ARN of the latest version of the definition.

type: string

400

invalid request

GeneralError (p. 476)

```json
{
    "Message": "string",
    "ErrorDetails": [
    {
        "DetailedErrorCode": "string",
        "DetailedErrorMessage": "string"
    }
    ]
}
```

GeneralError

General error information.

type: object

required: ["Message"]
CreateSubscriptionDefinitionVersion

Creates a version of a subscription definition which has already been defined.

URI: POST /greengrass/definition/subscriptions/SubscriptionDefinitionId/versions

Produce: application/json

CLI:

```bash
aws greengrass create-subscription-definition-version \
  --subscription-definition-id <value> \
  --subscriptions <value> \
  --amzn-client-token <value> \
  --cli-input-json <value> \
  --generate-cli-skeleton
```

cli-input-json format:

```json
{
  "SubscriptionDefinitionId": "string",
  "Subscriptions": [
    {
      "Id": "string",
      "Source": "string",
      "Subject": "string",
      "Target": "string"
    }
  ],
  "AmznClientToken": "string"
}
```
Parameters:

SubscriptionDefinitionId (p. 452)

The ID of the subscription definition.
where used: path; required: true
type: string

CreateSubscriptionDefinitionVersionRequestBody (p. 447)

where used: body; required: true

```json
{
   "Subscriptions": [
      {
         "Id": "string",
         "Source": "string",
         "Subject": "string",
         "Target": "string"
      }
   ]
}
```
schema: SubscriptionDefinitionVersion

Information about a subscription definition version.

type: object

Subscriptions

A list of subscriptions.

type: array

items: Subscription (p. 518)

Subscription

Information about a subscription.

type: object

required: ["Id", "Source", "Subject", "Target"]

Id

The id of the subscription.

type: string

Source

The source of the subscription. Can be a thing ARN, a Lambda function ARN, 'cloud' (which represents the AWS IoT cloud), or 'GGShadowService'.

type: string

Subject

The subject of the message.
type: string

Target
Where the message is sent to. Can be a thing ARN, a Lambda function ARN, 'cloud' (which represents the AWS IoT cloud), or 'GGShadowService'.

type: string

X-Amzn-Client-Token (p. 455)
A client token used to correlate requests and responses.

where used: header; required: false

type: string

Responses:

200 (CreateSubscriptionDefinitionVersionResponse)

VersionInformation (p. 521)

{
   "Arn": "string",
   "Id": "string",
   "Version": "string",
   "CreationTimestamp": "string"
}

VersionInformation
Information about a version.

   type: object

Arn
The ARN of the version.

type: string

Id
The ID of the version.

type: string

Version
The unique ID of the version.

type: string

CreationTimestamp
The time, in milliseconds since the epoch, when the version was created.

type: string

400
invalid request

GeneralError (p. 476)
DeleteCoreDefinition

Deletes a core definition.

**URI:** DELETE /greengrass/definition/cores/CoreDefinitionId

**Produces:** application/json

**CLI:**

```
aws greengrass delete-core-definition \
```
DeleteCoreDefinition

```
--core-definition-id <value>  \n[--cli-input-json <value>] \n[--generate-cli-skeleton]
```

cli-input-json format:

```
{
  "CoreDefinitionId": "string"
}
```

**Parameters:**

*CoreDefinitionId* (p. 430)

The ID of the core definition.

where used: path; required: true

type: string

**Responses:**

**200**

success

*Empty* (p. 467)

```
{
}
```

Empty Schema

Empty
type: object

**400**

invalid request

*GeneralError* (p. 476)

```
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

GeneralError

General error information.
type: object
DeleteDeviceDefinition

Deletes a device definition.

URI: DELETE /greengrass/definition/devices/{DeviceDefinitionId}

Produces: application/json

CLI:

```bash
aws greengrass delete-device-definition
   --device-definition-id <value>
   [--cli-input-json <value>] 
   [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
   "DeviceDefinitionId": "string"
}
```

Parameters:

**DeviceDefinitionId (p. 448)**

The ID of the device definition.
DeleteDeviceDefinition

where used: path; required: true

type: string

Responses:

200

success

Empty (p. 467)

{  }

Empty Schema

Empty
type: object

400

invalid request

GeneralError (p. 476)

{
   "Message": "string",
   "ErrorDetails": [
      {
         "DetailedErrorCode": "string",
         "DetailedErrorMessage": "string"
      }
   ]
}

GeneralError

General error information.
type: object
required: ["Message"]
Message
A message containing information about the error.
type: string
ErrorDetails
A list of error details.
type: array
items: ErrorDetail (p. 468)
ErrorDetail
Details about the error.
type: object
DetailedErrorCode
  A detailed error code.
type: string
DetailedErrorMessage
  A detailed error message.
type: string

DeleteFunctionDefinition

Deletes a Lambda function definition.

URI: DELETE /greengrass/definition/functions/FunctionDefinitionId

Produces: application/json

CLI:

```bash
aws greengrass delete-function-definition \\
  --function-definition-id <value> \\
  [--cli-input-json <value>] \\
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "FunctionDefinitionId": "string"
}
```

Parameters:

**FunctionDefinitionId** (p. 449)

The ID of the Lambda function definition.

  where used: path; required: true

  type: string

Responses:

**200**

success

Empty (p. 467)

```json
{
}
```
Empty Schema

Empty

type: object

400

Invalid request

**GeneralError (p. 476)**

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

**GeneralError**

General error information.

type: object

required: ["Message"]

**Message**

A message containing information about the error.

type: string

**ErrorDetails**

A list of error details.

type: array

items: **ErrorDetail (p. 468)**

**ErrorDetail**

Details about the error.

type: object

**DetailedErrorCode**

A detailed error code.

type: string

**DetailedErrorMessage**

A detailed error message.

type: string

---

**DeleteGroup**

Deletes a group.
DeleteGroup

URI: DELETE /greengrass/groups/GroupId

Produces: application/json

CLI:

```
aws greengrass delete-group
   --group-id <value> \
   [--cli-input-json <value>] \
   [--generate-cli-skeleton]
```

cli-input-json format:

```
{
   "GroupId": "string"
}
```

Parameters:

**GroupId (p. 449)**

The ID of the AWS Greengrass group.

where used: path; required: true
type: string

Responses:

**200**

success

Empty (p. 467)

```
{
}
```

Empty Schema

Empty
type: object

**400**

invalid request

GeneralError (p. 476)

```
{
   "Message": "string",
   "ErrorDetails": [
   {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
   }
   ]
}```
GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

DeleteLoggerDefinition

Deletes a logger definition.

URI: DELETE /greengrass/definition/loggers/LoggerDefinitionId

Produces: application/json

CLI:

```
aws greengrass delete-logger-definition \
  --logger-definition-id <value>  \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```
{
  "LoggerDefinitionId": "string"
}
```
Parameters:

**LoggerDefinitionId (p. 450)**

The ID of the logger definition.

where used: path; required: true
type: string

Responses:

200

success

Empty (p. 467)

```javascript
{}
```

Empty Schema

Empty
type: object

400

invalid request

**GeneralError (p. 476)**

```javascript
{
   "Message": "string",
   "ErrorDetails": [
       {
           "DetailedErrorCode": "string",
           "DetailedErrorMessage": "string"
       }
   ]
}
```

GeneralError

General error information.
type: object

Message

A message containing information about the error.
type: string

ErrorDetails

A list of error details.
DeleteResourceDefinition

Deletes a resource definition.

URI: DELETE /greengrass/definition/resources/ResourceDefinitionId

Produces: application/json

CLI:

```bash
aws greengrass delete-resource-definition \
  --resource-definition-id <value> \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "resource-definition-id": "string"
}
```

Parameters:

**ResourceDefinitionId (p. 452)**

The ID of the resource definition.

where used: path; required: true

type: string

Responses:

**200**

success
Empty (p. 467)

{}  

Empty Schema

Empty

type: object

400

invalid request

GeneralError (p. 476)

{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.
DeleteSubscriptionDefinition

Deletes a subscription definition.

URI: DELETE /greengrass/definition/subscriptions/SubscriptionDefinitionId

Produces: application/json

CLI:

```
aws greengrass delete-subscription-definition \
   --subscription-definition-id <value>  \
   [--cli-input-json <value>] \
   [--generate-cli-skeleton]
```

cli-input-json format:

```
{
    "SubscriptionDefinitionId": "string"
}
```

Parameters:

**SubscriptionDefinitionId** *(p. 452)*

The ID of the subscription definition.

where used: path; required: true

type: string

Responses:

**200**

success

Empty *(p. 467)*

```
{
}
```

Empty Schema

Empty

type: object

**400**

invalid request

GeneralError *(p. 476)*
DisassociateRoleFromGroup

Disassociates the role from a group.

**URI:** DELETE /greengrass/groups/GroupId/role

**CLI:**

```
aws greengrass disassociate-role-from-group \
  --group-id <value> \
  [--cli-input-json <value>] \ 
  [--generate-cli-skeleton]
```
cli-input-json format:

```json
{
  "GroupId": "string"
}
```

Parameters:

**GroupId** (p. 449)

The ID of the AWS Greengrass group.

- where used: path; required: true
- type: string

Responses:

**200**

success

**DisassociateRoleFromGroupResponse** (p. 467)

```json
{
  "DisassociatedAt": "string"
}
```

DisassociateRoleFromGroupResponse

- type: object
- **DisassociatedAt**
  - The time, in milliseconds since the epoch, when the role was disassociated from the group.
  - type: string

**400**

invalid request

**GeneralError** (p. 476)

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

GeneralError

- General error information.
- type: object
required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

500

server error

GeneralError (p. 476)

{
  "Message": "string",
  "ErrorDetails": [
  {
    "DetailedErrorCode": "string",
    "DetailedErrorMessage": "string"
  }
  ]
}

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.
DisassociateServiceRoleFromAccount

Disassociates the service role from your account. Without a service role, deployments will not work.

URI: DELETE /greengrass/servicerole

CLI:

```
aws greengrass disassociate-service-role-from-account  
[--cli-input-json <value>] 
[--generate-cli-skeleton]
```

Responses:

200

success

DisassociateServiceRoleFromAccountResponse

```
{  
  "DisassociatedAt": "string"  
}
```

DisassociateServiceRoleFromAccountResponse

- type: object
- DisassociatedAt

- The time when the service role was disassociated from the account.

500

server error

GeneralError
GetAssociatedRole

Retrieves the role associated with a particular group.

URI: GET /greengrass/groups/GroupId/role

CLI:

```
aws greengrass get-associated-role \
  --group-id <value> \
  [--cli-input-json <value>] \
```
cli-input-json format:

```json
{
  "GroupId": "string"
}
```

### Parameters:

**GroupId (p. 449)**

The ID of the AWS Greengrass group.

where used: path; required: true
type: string

### Responses:

**200**

success

GetAssociatedRoleResponse (p. 477)

```json
{
  "AssociatedAt": "string",
  "RoleArn": "string"
}
```

GetAssociatedRoleResponse
type: object

**AssociatedAt**

The time when the role was associated with the group.
type: string

**RoleArn**

The ARN of the role that is associated with the group.
type: string

**400**

invalid request

GeneralError (p. 476)

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```
GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

500

server error

GeneralError (p. 476)
required: ["Message"]

**Message**

A message containing information about the error.

type: string

**ErrorDetails**

A list of error details.

type: array

items: ErrorDetail (p. 468)

**ErrorDetail**

Details about the error.

type: object

**DetailedErrorCode**

A detailed error code.

type: string

**DetailedErrorMessage**

A detailed error message.

type: string

---

**GetConnectivityInfo**

Retrieves the connectivity information for a core.

**URI:** GET /greengrass/things/ThingName/connectivityInfo

**Produces:** application/json

**CLI:**

```bash
aws greengrass get-connectivity-info
   --thing-name <value>
   [--cli-input-json <value>]
   [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
   "ThingName": "string"
}
```

**Parameters:**

**ThingName (p. 453)**

The thing name.
where used: path; required: true
type: string

Responses:

200

success

GetConnectivityInfoResponse (p. 477)

{
  "message": "string",
  "ConnectivityInfo": [
    {
      "Id": "string",
      "HostAddress": "string",
      "PortNumber": 0x01,
      "Metadata": "string"
    }
  ]
}

Information about a Greengrass core's connectivity.

message

A message about the connectivity info request.
type: string

ConnectivityInfo

Connectivity info list.
type: array
items: ConnectivityInfo (p. 458)

Information about a Greengrass core's connectivity.

required: ["Id", "HostAddress"]

Id

The ID of the connectivity information.
type: string

HostAddress

The endpoint for the Greengrass core. Can be an IP address or DNS.
type: string

PortNumber

The port of the Greengrass core. Usually 8883.
type: integer
format: int32
Metadata

Metadata for this endpoint.

type: string

400

invalid request

GeneralError (p. 476)

{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

500

server error

GeneralError (p. 476)
GetCoreDefinition

Retrieves information about a core definition version.

URI: GET /greengrass/definition/cores/CoreDefinitionId

Produces: application/json

CLI:

```bash
aws greengrass get-core-definition \
    --core-definition-id <value> \
    [--cli-input-json <value>] \
```
[--generate-cli-skeleton]

cli-input-json format:

```json
{
  "CoreDefinitionId": "string"
}
```

**Parameters:**

**CoreDefinitionId (p. 430)**

The ID of the core definition.

where used: path; required: true
type: string

**Responses:**

**200 (GetCoreDefinitionResponse)**

**DefinitionInformation (p. 463)**

```json
{
  "Name": "string",
  "Id": "string",
  "Arn": "string",
  "LastUpdatedTimestamp": "string",
  "CreationTimestamp": "string",
  "LatestVersion": "string",
  "LatestVersionArn": "string"
}
```

**DefinitionInformation**

Information about a definition.
type: object

**Name**

The name of the definition.
type: string

**Id**

The ID of the definition.
type: string

**Arn**

The ARN of the definition.
type: string

**LastUpdatedTimestamp**

The time, in milliseconds since the epoch, when the definition was last updated.
type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the definition was created.

type: string

LatestVersion

The latest version of the definition.

type: string

LatestVersionArn

The ARN of the latest version of the definition.

type: string

400

invalid request

GeneralError (p. 476)

```json
{
    "Message": "string",
    "ErrorDetails": [
    {
        "DetailedErrorCode": "string",
        "DetailedErrorMessage": "string"
    }
    ]
}
```

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.
GetCoreDefinitionVersion

Retrieves information about a core definition version.

**URI:** GET /greengrass/definition/cores/CoreDefinitionId/versions/CoreDefinitionVersionId

**Produces:** application/json

**CLI:**

```
aws greengrass get-core-definition-version
   --core-definition-id <value> \
   --core-definition-version-id <value> \n   [--cli-input-json <value>] \n   [--generate-cli-skeleton]
```

**Parameters:**

- **CoreDefinitionId (p. 430)**
  - The ID of the core definition.
    - where used: path; required: true
    - type: string

- **CoreDefinitionVersionId (p. 430)**
  - The ID of the core definition version.
    - where used: path; required: true
    - type: string

**Responses:**

- **200**
  - success
    - GetCoreDefinitionVersionResponse (p. 478)
GetCoreDefinitionVersion

```
{
    "Arn": "string",
    "Id": "string",
    "Version": "string",
    "CreationTimestamp": "string",
    "Definition": {
        "Cores": [
            {
                "Id": "string",
                "ThingArn": "string",
                "CertificateArn": "string",
                "SyncShadow": true
            }
        ]
    }
}
```

GetCoreDefinitionVersionResponse

type: object

Arn

The ARN of the core definition version.

type: string

Id

The ID of the core definition version.

type: string

Version

The version of the core definition version.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the core definition version was created.

type: string

Definition

Information about a core definition version.

type: object

Cores

A list of cores in the core definition version.

type: array

items: Core (p. 458)

Core

Information about a core.

type: object

required: ["Id", "ThingArn", "CertificateArn"]
Id

The ID of the core.

type: string

ThingArn

The ARN of the thing which is the core.

type: string

CertificateArn

The ARN of the certificate associated with the core.

type: string

SyncShadow

If true, the core's local shadow is automatically synced with the cloud.

type: boolean

400

invalid request

GeneralError (p. 476)

```json
{
   "Message": "string",
   "ErrorDetails": [
      {
         "DetailedErrorCode": "string",
         "DetailedErrorMessage": "string"
      }
   ]
}
```

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object
GetDeploymentStatus

Returns the status of a deployment.

URI: GET /greengrass/groups/GroupId/deployments/DeploymentId/status

Produces: application/json

CLI:

```bash
aws greengrass get-deployment-status \  
   --group-id <value> \  
   --deployment-id <value> \  
   [--cli-input-json <value>] \  
   [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
   "GroupId": "string",
   "DeploymentId": "string"
}
```

Parameters:

**GroupId** *(p. 449)*

The ID of the AWS Greengrass group.

where used: path; required: true

type: string

**DeploymentId** *(p. 448)*

The ID of the deployment.

where used: path; required: true

type: string

Responses:

**200**

Success. The response body contains the status of the deployment for the group.
GetDeploymentStatusResponse (p. 480)

```json
{
    "DeploymentStatus": "string",
    "DeploymentType": "NewDeployment|Redeployment|ResetDeployment|ForceResetDeployment",
    "UpdatedAt": "string",
    "ErrorMessage": "string",
    "ErrorDetails": [
        {
            "DetailedErrorCode": "string",
            "DetailedErrorMessage": "string"
        }
    ]
}
```

GetDeploymentStatusResponse

Information about the status of a deployment for a group.

**type:** object

**DeploymentStatus**

The status of the deployment.

**type:** string

**DeploymentType**

**type:** string

**enum:** ["NewDeployment", "Redeployment", "ResetDeployment", "ForceResetDeployment"]

**UpdatedAt**

The time, in milliseconds since the epoch, when the deployment status was updated.

**type:** string

**ErrorMessage**

Error message

**type:** string

**ErrorDetails**

A list of error details.

**type:** array

**items:** [ErrorDetail (p. 468)]

**ErrorDetail**

Details about the error.

**type:** object

**DetailedErrorCode**

A detailed error code.

**type:** string

**DetailedErrorMessage**

A detailed error message.
GetDeviceDefinition

Retrieves information about a device definition.

URI: GET /greengrass/definition/devices/DeviceDefinitionId
**CLI:**

```bash
greengrass get-device-definition \
  --device-definition-id <value> \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "DeviceDefinitionId": "string"
}
```

**Parameters:**

**DeviceDefinitionId** *(p. 448)*

The ID of the device definition.

- where used: path; required: true
- type: string

**Responses:**

**200 (GetDeviceDefinitionResponse)**

**DefinitionInformation** *(p. 463)*

```json
{
  "Name": "string",
  "Id": "string",
  "Arn": "string",
  "LastUpdatedTimestamp": "string",
  "CreationTimestamp": "string",
  "LatestVersion": "string",
  "LatestVersionArn": "string"
}
```

**DefinitionInformation**

Information about a definition.

- type: object

**Name**

The name of the definition.

- type: string

**Id**

The ID of the definition.

- type: string
Arn

The ARN of the definition.

type: string

LastUpdatedTimestamp

The time, in milliseconds since the epoch, when the definition was last updated.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the definition was created.

type: string

LatestVersion

The latest version of the definition.

type: string

LatestVersionArn

The ARN of the latest version of the definition.

type: string

400

invalid request

GeneralError (p. 476)

```
{
    "Message": "string",
    "ErrorDetails": [
        {
            "DetailedErrorCode": "string",
            "DetailedErrorMessage": "string"
        }
    ]
}
```

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)
ErrorDetail
  Details about the error.
  type: object
DetailedErrorCode
  A detailed error code.
  type: string
DetailedErrorMessage
  A detailed error message.
  type: string

GetDeviceDefinitionVersion
Retrieves information about a device definition version.

URI: GET /greengrass/definition/devices/DeviceDefinitionId/versions/DeviceDefinitionVersionId

Produces: application/json

CLI:

```
aws greengrass get-device-definition-version \
  --device-definition-id <value> \
  --device-definition-version-id <value> \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```
{
  "DeviceDefinitionId": "string",
  "DeviceDefinitionVersionId": "string"
}
```

Parameters:

DeviceDefinitionId (p. 448)
  The ID of the device definition.
  where used: path; required: true
  type: string

DeviceDefinitionVersionId (p. 448)
  The ID of the device definition version.
  where used: path; required: true
  type: string
Responses:

200 (GetDeviceDefinitionVersionResponse)

GetDeviceDefinitionVersionResponse (p. 481)

```json
{
  "Arn": "string",
  "Id": "string",
  "Version": "string",
  "CreationTimestamp": "string",
  "Definition": {
    "Devices": [
      {
        "Id": "string",
        "ThingArn": "string",
        "CertificateArn": "string",
        "SyncShadow": true
      }
    ]
  }
}
```

GetDeviceDefinitionVersionResponse

type: object

Arn

The ARN of the device definition version.

type: string

Id

The ID of the device definition version.

type: string

Version

The version of the device definition version.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the device definition version was created.

type: string

Definition

Information about a device definition version.

  type: object

Devices

A list of devices in the definition version.

  type: array

  items: Device (p. 465)
Device
   Information about a device.
   type: object
   required: ["Id", "ThingArn", "CertificateArn"]

Id
   The ID of the device.
   type: string

ThingArn
   The thing ARN of the device.
   type: string

CertificateArn
   The ARN of the certificate associated with the device.
   type: string

SyncShadow
   If true, the device's local shadow will be automatically synced with the cloud.
   type: boolean

400
invalid request

GeneralError (p. 476)

{  
   "Message": "string",
   "ErrorDetails": [  
      {  
         "DetailedErrorCode": "string",
         "DetailedErrorMessage": "string"
      }
   ]
}

GeneralError
   General error information.
   type: object
   required: ["Message"]

Message
   A message containing information about the error.
   type: string

ErrorDetails
   A list of error details.
   type: array
ErrorDetail (p. 468)
ErrorDetail
   Details about the error.
   type: object
DetailedErrorCode
   A detailed error code.
   type: string
DetailedErrorMessage
   A detailed error message.
   type: string

GetFunctionDefinition
Retrieves information about a Lambda function definition, including its creation time and latest version.

URL: GET /greengrass/definition/functions/FunctionDefinitionId

Produces: application/json

CLI:

```bash
aws greengrass get-function-definition \
   --function-definition-id <value>  \
   [--cli-input-json <value>] \
   [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
   "FunctionDefinitionId": "string"
}
```

Parameters:

FunctionDefinitionId (p. 449)
The ID of the Lambda function definition.
   where used: path; required: true
type: string

Responses:

200 (GetFunctionDefinitionResponse)
success
   DefinitionInformation (p. 463)


```json
{
    "Name": "string",
    "Id": "string",
    "Arn": "string",
    "LastUpdatedTimestamp": "string",
    "CreationTimestamp": "string",
    "LatestVersion": "string",
    "LatestVersionArn": "string"
}
```

DefinitionInformation

Information about a definition.

type: object

Name

The name of the definition.

type: string

Id

The ID of the definition.

type: string

Arn

The ARN of the definition.

type: string

LastUpdatedTimestamp

The time, in milliseconds since the epoch, when the definition was last updated.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the definition was created.

type: string

LatestVersion

The latest version of the definition.

type: string

LatestVersionArn

The ARN of the latest version of the definition.

type: string

400

invalid request

GeneralError (p. 476)

```json
{
    "Message": "string",
    "ErrorDetails": [
```
GetFunctionDefinitionVersion

Retrieves information about a Lambda function definition version, including which Lambda functions are included in the version and their configurations.

URI: GET /greengrass/definition/functions/FunctionDefinitionId/versions/FunctionDefinitionVersionId

Produces: application/json

CLI:

```
aws greengrass get-function-definition-version \
  --function-definition-id <value> \
  --function-definition-version-id <value> \
  [--cli-input-json <value>] \
```
GetFunctionDefinitionVersion

cli-input-json format:

```json
{
    "FunctionDefinitionId": "string",
    "FunctionDefinitionVersionId": "string"
}
```

Parameters:

**FunctionDefinitionId** (p. 449)

The ID of the Lambda function definition.

where used: path; required: true

type: string

**FunctionDefinitionVersionId** (p. 449)

The ID of the function definition version.

where used: path; required: true

type: string

Responses:

**200**

success

**GetFunctionDefinitionVersionResponse** (p. 482)

```json
{
    "Arn": "string",
    "Id": "string",
    "Version": "string",
    "CreationTimestamp": "string",
    "Definition": {
        "Functions": [
            {
                "Id": "string",
                "FunctionArn": "string",
                "FunctionConfiguration": {
                    "Pinned": true,
                    "Executable": "string",
                    "ExecArgs": "string",
                    "MemorySize": 0,
                    "Timeout": 0,
                    "EncodingType": "binary|json",
                    "Environment": {
                        "Variables": {
                            "additionalProperty0": "string",
                            "additionalProperty1": "string",
                            "additionalProperty2": "string"
                        }
                    }
                },
                "ResourceAccessPolicies": [
                    {
                    }
                ]
            }
        ]
    }
}
```
GetFunctionDefinitionVersion

Information about a function definition version.

type: object

Arn

The ARN of the function definition version.

type: string

Id

The ID of the function definition version.

type: string

Version

The version of the function definition version.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the function definition version was created.

type: string

Definition

Information about a function definition version.

type: object

Functions

A list of Lambda functions in this function definition version.

type: array

items: Function (p. 469)

Information about a Lambda function.

type: object

required: ["Id"]

Id

The ID of the Lambda function.

type: string
FunctionArn
   The ARN of the Lambda function.
   type: string

FunctionConfiguration
   The configuration of the Lambda function.
   type: object

Pinned
   True if the function is pinned. Pinned means the function is long-lived and starts when the core starts.
   type: boolean

Executable
   The name of the function executable.
   type: string

ExecArgs
   The execution arguments.
   type: string

MemorySize
   The memory size, in KB, which the function requires.
   type: integer

Timeout
   The allowed function execution time, after which Lambda should terminate the function. This timeout still applies to pinned lambdas for each request.
   type: integer

EncodingType
   The expected encoding type of the input payload for the function. The default is json.
   type: string
   enum: ["binary", "json"]

Environment
   The environment configuration of the function.
   type: object

Variables
   Environment variables for the Lambda function's configuration.
   type: object
   additionalProperties: "type"=>"string"

ResourceAccessPolicies
   A list of the resources, with their permissions, to which the Lambda function will be granted access. A Lambda function can have at most 10 resources.
type: array

items: `ResourceAccessPolicy` (p. 511)

**ResourceAccessPolicy**

A policy used by the function to access a resource.

type: object

required: ["ResourceId"]

**ResourceId**

The ID of the resource. (This ID is assigned to the resource when you create the resource definition.)

type: string

**Permission**

The type of permission a function has to access a resource.

type: string

enum: ["ro", "rw"]

**AccessSysfs**

If true, the Lambda function is allowed to access the host's /sys folder. Use this when the Lambda function needs to read device information from /sys.

type: boolean

400

invalid request

**GeneralError** (p. 476)

```json
{
   "Message": "string",
   "ErrorDetails": [
       {
           "DetailedErrorCode": "string",
           "DetailedErrorMessage": "string"
       }
   ]
}
```

**GeneralError**

General error information.

type: object

required: ["Message"]

**Message**

A message containing information about the error.

type: string

**ErrorDetails**

A list of error details.
GetGroup

Retrieves information about a group.

URI: GET /greengrass/groups/{GroupId}

Produces: application/json

**CLI:**

```bash
aws greengrass get-group
  --group-id <value> 
  [--cli-input-json <value>] 
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "GroupId": "string"
}
```

**Parameters:**

**GroupId (p. 449)**

The ID of the AWS Greengrass group.

where used: path; required: true

type: string

**Responses:**

**200 (GetGroupResponse)**

success
DefinitionInformation (p. 463)

```json
{
  "Name": "string",
  "Id": "string",
  "Arn": "string",
  "LastUpdatedTimestamp": "string",
  "CreationTimestamp": "string",
  "LatestVersion": "string",
  "LatestVersionArn": "string"
}
```

**DefinitionInformation**

Information about a definition.

- **type**: object

**Name**

The name of the definition.

- **type**: string

**Id**

The ID of the definition.

- **type**: string

**Arn**

The ARN of the definition.

- **type**: string

**LastUpdatedTimestamp**

The time, in milliseconds since the epoch, when the definition was last updated.

- **type**: string

**CreationTimestamp**

The time, in milliseconds since the epoch, when the definition was created.

- **type**: string

**LatestVersion**

The latest version of the definition.

- **type**: string

**LatestVersionArn**

The ARN of the latest version of the definition.

- **type**: string

**400**

Invalid request

**GeneralError (p. 476)**

```json
{
```

119
GetGroupCertificateAuthority

Retreives the CA associated with a group. Returns the public key of the CA.

URI: GET /greengrass/groups/GroupId/certificateauthorities/CertificateAuthorityId

Produces: application/json

CLI:

aws greengrass get-group-certificate-authority \ --group-id <value> \
GetGroupCertificateAuthority

```
--certificate-authority-id <value> \n[--cli-input-json <value>] \n[--generate-cli-skeleton]
```

cli-input-json format:

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

**Parameters:**

**GroupId (p. 449)**

The ID of the AWS Greengrass group.

where used: path; required: true
type: string

**CertificateAuthorityId (p. 430)**

The ID of the certificate authority.

where used: path; required: true
type: string

**Responses:**

**200**

Success. The response body contains the PKI Configuration.

GetGroupCertificateAuthorityResponse (p. 485)

```
{
  "PemEncodedCertificate": "string",
  "GroupCertificateAuthorityArn": "string",
  "GroupCertificateAuthorityId": "string"
}
```

GetGroupCertificateAuthorityResponse

Information about a certificate authority for a group.
type: object

**PemEncodedCertificate**

The PEM encoded certificate for the group.
type: string

**GroupCertificateAuthorityArn**

The ARN of the certificate authority for the group.
type: string
GroupCertificateAuthorityId

The ID of the certificate authority for the group.

type: string

400

invalid request

**GeneralError** *(p. 476)*

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

**GeneralError**

General error information.

type: object

required: ["Message"]

**Message**

A message containing information about the error.

type: string

**ErrorDetails**

A list of error details.

type: array

items: **ErrorDetail** *(p. 468)*

**ErrorDetail**

Details about the error.

type: object

**DetailedErrorCode**

A detailed error code.

type: string

**DetailedErrorMessage**

A detailed error message.

type: string

500

server error

**GeneralError** *(p. 476)*
GetGroupCertificateConfiguration

Retrieves the current configuration for the CA used by the group.

URI: GET /greengrass/groups/GroupId/certificateauthorities/configuration/expiry

Produces: application/json

CLI:

```
aws greengrass get-group-certificate-configuration \
   --group-id <value>
```
GetGroupCertificateConfiguration

[--cli-input-json <value>] \
[--generate-cli-skeleton]

cli-input-json format:

```json
{
  "GroupId": "string"
}
```

Parameters:

**GroupId** (p. 449)

The ID of the AWS Greengrass group.

where used: path; required: true

type: string

Responses:

**200**

Success. The response body contains the PKI Configuration.

**GroupCertificateConfiguration** (p. 495)

```json
{
  "GroupId": "string",
  "CertificateAuthorityExpiryInMilliseconds": "string",
  "CertificateExpiryInMilliseconds": "string"
}
```

GroupCertificateConfiguration

Information about a group certificate configuration.

type: object

GroupId

The ID of the group certificate configuration.

type: string

CertificateAuthorityExpiryInMilliseconds

The amount of time remaining before the certificate authority expires, in milliseconds.

type: string

CertificateExpiryInMilliseconds

The amount of time remaining before the certificate expires, in milliseconds.

type: string

**400**

invalid request
GeneralError (p. 476)

```
{
    "Message": "string",
    "ErrorDetails": [
    {
        "DetailedErrorCode": "string",
        "DetailedErrorMessage": "string"
    }
    ]
}
```

**GeneralError**

General error information.

type: object

required: ["Message"]

**Message**

A message containing information about the error.

type: string

**ErrorDetails**

A list of error details.

type: array

items: [ErrorDetail (p. 468)]

**ErrorDetail**

Details about the error.

type: object

**DetailedErrorCode**

A detailed error code.

type: string

**DetailedErrorMessage**

A detailed error message.

type: string

500

server error

GeneralError (p. 476)

```
{
    "Message": "string",
    "ErrorDetails": [
    {
        "DetailedErrorCode": "string",
        "DetailedErrorMessage": "string"
    }
    ]
}
```
GetGroupVersion

Retrieves information about a group version.

URI: GET /greengrass/groups/<GroupId>/versions/<GroupVersionId>

Produces: application/json

CLI:

```bash
aws greengrass get-group-version \
  --group-id <value> \
  --group-version-id <value> \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
    "GroupId": "string",
```
"GroupVersionId": "string"
}

Parameters:

GroupId (p. 449)

The ID of the AWS Greengrass group.

where used: path; required: true
type: string

GroupVersionId (p. 450)

The ID of the group version.

where used: path; required: true
type: string

Responses:

200

success

GetGroupVersionResponse (p. 486)

{
  "Arn": "string",
  "Id": "string",
  "Version": "string",
  "CreationTimestamp": "string",
  "Definition": {
    "CoreDefinitionVersionArn": "string",
    "DeviceDefinitionVersionArn": "string",
    "FunctionDefinitionVersionArn": "string",
    "SubscriptionDefinitionVersionArn": "string",
    "LoggerDefinitionVersionArn": "string",
    "ResourceDefinitionVersionArn": "string"
  }
}

GetGroupVersionResponse

Information about a group version.
type: object

Arn

The ARN of the group version.
type: string

Id

The ID of the group version.
type: string
Version
   The unique ID for the version of the group.
   type: string
CreationTimestamp
   The time, in milliseconds since the epoch, when the group version was created.
   type: string
Definition
   Information about a group version.
   type: object
CoreDefinitionVersionArn
   The ARN of the core definition version for this group.
   type: string
DeviceDefinitionVersionArn
   The ARN of the device definition version for this group.
   type: string
FunctionDefinitionVersionArn
   The ARN of the function definition version for this group.
   type: string
SubscriptionDefinitionVersionArn
   The ARN of the subscription definition version for this group.
   type: string
LoggerDefinitionVersionArn
   The ARN of the logger definition version for this group.
   type: string
ResourceDefinitionVersionArn
   The resource definition version ARN for this group.
   type: string

400
invalid request

GeneralError (p. 476)

```json
{
   "Message": "string",
   "ErrorDetails": [
   {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
   }
   ]
}```
GetLoggerDefinition

Retrieves information about a logger definition.

URI: GET /greengrass/definition/loggers/LoggerDefinitionId

Produces: application/json

CLI:

```
aws greengrass get-logger-definition \
  --logger-definition-id <value> \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```
{
  "LoggerDefinitionId": "string"
}
Parameters:

**LoggerDefinitionId (p. 450)**

The ID of the logger definition.

where used: path; required: true
type: string

Responses:

**200 (GetLoggerDefinitionResponse)**

**DefinitionInformation (p. 463)**

```
{
    "Name": "string",
    "Id": "string",
    "Arn": "string",
    "LastUpdatedTimestamp": "string",
    "CreationTimestamp": "string",
    "LatestVersion": "string",
    "LatestVersionArn": "string"
}
```

**DefinitionInformation**

Information about a definition.
type: object

**Name**

The name of the definition.
type: string

**Id**

The ID of the definition.
type: string

**Arn**

The ARN of the definition.
type: string

**LastUpdatedTimestamp**

The time, in milliseconds since the epoch, when the definition was last updated.
type: string

**CreationTimestamp**

The time, in milliseconds since the epoch, when the definition was created.
type: string
LatestVersion
The latest version of the definition.
   type: string

LatestVersionArn
The ARN of the latest version of the definition.
   type: string

400
invalid request

GeneralError (p. 476)

```json
{
   "Message": "string",
   "ErrorDetails": [
   {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
   }
   ]
}
```

GeneralError
General error information.
   type: object
   required: ["Message"]

Message
A message containing information about the error.
   type: string

ErrorDetails
A list of error details.
   type: array
   items: ErrorDetail (p. 468)

ErrorDetail
Details about the error.
   type: object

DetailedErrorCode
A detailed error code.
   type: string

DetailedErrorMessage
A detailed error message.
   type: string
GetLoggerDefinitionVersion

Retrieves information about a logger definition version.

URI: GET /greengrass/definition/loggers/LoggerDefinitionId/
versions/LoggerDefinitionVersionId

Produces: application/json

CLI:

```
aws greengrass get-logger-definition-version \
   --logger-definition-version-id <value> \
   --logger-definition-id <value> \
   [--cli-input-json <value>] \
   [--generate-cli-skeleton]
```

cli-input-json format:

```
{
  "LoggerDefinitionVersionId": "string",
  "LoggerDefinitionId": "string"
}
```

Parameters:

**LoggerDefinitionVersionId** (p. 450)

The ID of the logger definition version.

where used: path; required: true

type: string

**LoggerDefinitionId** (p. 450)

The ID of the logger definition.

where used: path; required: true

type: string

Responses:

**200** (GetLoggerDefinitionVersionResponse)

  success

  GetLoggerDefinitionVersionResponse (p. 487)

```
{
  "Arn": "string",
  "Id": "string",
  "Version": "string",
  "CreationTimestamp": "string",
  "Definition": {
    "Loggers": [ ...
```
GetLoggerDefinitionVersion

GetLoggerDefinitionVersionResponse

Information about a logger definition version.

type: object

Arn

The ARN of the logger definition version.

type: string

Id

The ID of the logger definition version.

type: string

Version

The version of the logger definition version.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the logger definition version was created.

type: string

Definition

Information about a logger definition version.

type: object

Loggers

A list of loggers.

type: array

items: Logger (p. 505)

Logger

Information about a logger

type: object

required: ["Id", "Type", "Component", "Level"]

Id

The id of the logger.

type: string
Type
  type: string
eenum: ["FileSystem", "AWSCloudWatch"]
Component
  type: string
eenum: ["GreengrassSystem", "Lambda"]
Level
  type: string
eenum: ["DEBUG", "INFO", "WARN", "ERROR", "FATAL"]
Space
  The amount of file space, in KB, to use if the local file system is used for logging purposes.
type: integer

400
invalid request

GeneralError (p. 476)

```json
{
  "Message": "string",
  "ErrorDetails": [  
    {    
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }  
  ]
}
```

GeneralError
  General error information.
type: object
required: ["Message"]
Message
  A message containing information about the error.
type: string
ErrorDetails
  A list of error details.
type: array
items: ErrorDetail (p. 468)
ErrorDetail
  Details about the error.
type: object
GetResourceDefinition

Retrieves information about a resource definition, including its creation time and latest version.

URI: GET /greengrass/definition/resources/{ResourceDefinitionId}

Produces: application/json

CLI:

```bash
aws greengrass get-resource-definition \
   --resource-definition-id <value> \
   [--cli-input-json <value>] \
   [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
   "ResourceDefinitionId": "string"
}
```

Parameters:

**ResourceDefinitionId** (p. 452)

The ID of the resource definition.

where used: path; required: true

type: string

Responses:

**200 (GetResourceDefinitionResponse)**

success

**DefinitionInformation** (p. 463)

```json
{
   "Name": "string",
   "Id": "string",
   "Arn": "string",
   "LastUpdatedTimestamp": "string",
}
```
DefinitionInformation

Information about a definition.

type: object

Name

The name of the definition.

type: string

Id

The ID of the definition.

type: string

Arn

The ARN of the definition.

type: string

LastUpdatedTimestamp

The time, in milliseconds since the epoch, when the definition was last updated.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the definition was created.

type: string

LatestVersion

The latest version of the definition.

type: string

LatestVersionArn

The ARN of the latest version of the definition.

type: string

400

invalid request

GeneralError (p. 476)
GetResourceDefinitionVersion

Retrieves information about a resource definition version, including which resources are included in the version.

URI: GET /greengrass/definition/resources/ResourceDefinitionId/versions/ResourceDefinitionVersionId

Produces: application/json

CLI:

```
aws greengrass get-resource-definition-version \
  --resource-definition-id <value> \
  --resource-definition-version-id <value> \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:
GetResourceDefinitionVersion

```json
{
    "ResourceDefinitionId": "string",
    "ResourceDefinitionVersionId": "string"
}
```

**Parameters:**

**ResourceDefinitionId (p. 452)**

The ID of the resource definition.

- where used: path; required: true
- type: string

**ResourceDefinitionVersionId (p. 452)**

The ID of the resource definition version.

- where used: path; required: true
- type: string

**Responses:**

**200**

success

GetResourceDefinitionVersionResponse (p. 489)

```json
{
    "Arn": "string",
    "Id": "string",
    "Version": "string",
    "CreationTimestamp": "string",
    "Definition": {
        "Resources": [
            {
                "Id": "string",
                "Name": "string",
                "ResourceDataContainer": {
                    "LocalDeviceResourceData": {
                        "SourcePath": "string",
                        "GroupOwnerSetting": {
                            "AutoAddGroupOwner": true,
                            "GroupOwner": "string"
                        }
                    },
                    "LocalVolumeResourceData": {
                        "SourcePath": "string",
                        "DestinationPath": "string",
                        "GroupOwnerSetting": {
                            "AutoAddGroupOwner": true,
                            "GroupOwner": "string"
                        }
                    }
                },
                "SageMakerMachineLearningModelResourceData": {
                    "SageMakerJobArn": "string",
                    "DestinationPath": "string"
                }
            }
        ]
    }
}
```
GetResourceDefinitionVersion

Information about a resource definition version.

type: object

Arn

Arn of the resource definition version.

type: string

Id

The ID of the resource definition version.

type: string

Version

The version of the resource definition version.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the resource definition version was created.

type: string

Definition

Information about a resource definition version.

type: object

Resources

A list of resources.

type: array

items: Resource (p. 508)

Resource

Information about a resource.

type: object

required: ["Id", "Name", "ResourceDataContainer"]

Id

The resource ID, used to refer to a resource in the Lambda function configuration. Max length is 128 characters with pattern [a-zA-Z0-9:_-]+. This must be unique within a Greengrass group.
type: string
Name

The descriptive resource name, which is displayed on the AWS Greengrass console. Max length 128 characters with pattern \[a-zA-Z0-9_:\-_]+. This must be unique within a Greengrass group.

type: string
ResourceDataContainer

A container for resource data. The container takes only one of the following supported resource data types: LocalDeviceResourceData, LocalVolumeResourceData, SageMakerMachineLearningModelResourceData, S3MachineLearningModelResourceData.

type: object
NOTE: oneOf
LocalDeviceResourceData

Attributes that define a local device resource.

type: object
SourcePath

The local absolute path of the device resource. The source path for a device resource can refer only to a character device or block device under /dev.

type: string
GroupOwnerSetting

Group owner related settings for local resources.

type: object
AutoAddGroupOwner

If true, AWS Greengrass automatically adds the specified Linux OS group owner of the resource to the Lambda process privileges. Thus the Lambda process will have the file access permissions of the added Linux group.

type: boolean
GroupOwner

The name of the Linux OS group whose privileges will be added to the Lambda process. This field is optional.

type: string
LocalVolumeResourceData

Attributes that define a local volume resource.

type: object
SourcePath

The local absolute path of the volume resource on the host. The source path for a volume resource type cannot start with /proc or /sys.

type: string
DestinationPath

The absolute local path of the resource inside the Lambda environment.

type: string

GroupOwnerSetting

Group owner related settings for local resources.

type: object

AutoAddGroupOwner

If true, AWS Greengrass automatically adds the specified Linux OS group owner of the resource to the Lambda process privileges. Thus the Lambda process will have the file access permissions of the added Linux group.

type: boolean

GroupOwner

The name of the Linux OS group whose privileges will be added to the Lambda process. This field is optional.

type: string

SageMakerMachineLearningModelResourceData

Attributes that define an Amazon SageMaker machine learning resource.

type: object

SageMakerJobArn

The ARN of the Amazon SageMaker training job that represents the source model.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.

type: string

S3MachineLearningModelResourceData

Attributes that define an Amazon S3 machine learning resource.

type: object

S3Uri

The URI of the source model in an Amazon S3 bucket. The model package must be in tar.gz or .zip format.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.

type: string

400

invalid request

GeneralError (p. 476)
GetServiceRoleForAccount

Retrieves the service role that is attached to your account.

URI: GET /greengrass/servicerole

CLI:

```
aws greengrass get-service-role-for-account \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```
Responses:

200

success

GetServiceRoleForAccountResponse (p. 492)

```
{
  "AssociatedAt": "string",
  "RoleArn": "string"
}
```

GetServiceRoleForAccountResponse

    type: object

AssociatedAt

The time when the service role was associated with the account.

    type: string

RoleArn

The ARN of the role which is associated with the account.

    type: string

500

server error

GeneralError (p. 476)

```
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

GeneralError

    General error information.

    type: object

required: ["Message"]

Message

A message containing information about the error.

    type: string

ErrorDetails

A list of error details.

    type: array
GetSubscriptionDefinition

Retrieves information about a subscription definition.

URI: GET /greengrass/definition/subscriptions/SubscriptionDefinitionId

Produces: application/json

CLI:

```
aws greengrass get-subscription-definition \
   --subscription-definition-id <value> \
   [--cli-input-json <value>] \
   [--generate-cli-skeleton]
```

cli-input-json format:

```
{
   "SubscriptionDefinitionId": "string"
}
```

Parameters:

**SubscriptionDefinitionId** *(p. 452)*

The ID of the subscription definition.

where used: path; required: true

type: string

Responses:

**200** *(GetSubscriptionDefinitionResponse)*

```
DefinitionInformation (p. 463)
```

```
"Name": "string",
"Id": "string",
"Arn": "string",
"LastUpdatedTimestamp": "string",
"CreationTimestamp": "string",
"LatestVersion": "string",
"LatestVersionArn": "string"
}

DefinitionInformation

Information about a definition.

type: object

Name

The name of the definition.

type: string

Id

The ID of the definition.

type: string

Arn

The ARN of the definition.

type: string

LastUpdatedTimestamp

The time, in milliseconds since the epoch, when the definition was last updated.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the definition was created.

type: string

LatestVersion

The latest version of the definition.

type: string

LatestVersionArn

The ARN of the latest version of the definition.

type: string

400

invalid request

GeneralError (p. 476)
GetSubscriptionDefinitionVersion

Retrieves information about a subscription definition version.

**URI:** GET /greengrass/definition/subscriptions/\{SubscriptionDefinitionId\}/versions/\{SubscriptionDefinitionVersionId\}

**Produces:** application/json

**CLI:**

```
aws greengrass get-subscription-definition-version \  --subscription-definition-id <value> \  --subscription-definition-version-id <value> \  [--cli-input-json <value>] \  [--generate-cli-skeleton]
```

cli-input-json format:
GetSubscriptionDefinitionVersion

```
{
"SubscriptionDefinitionId": "string",
"SubscriptionDefinitionVersionId": "string"
}
```

**Parameters:**

**SubscriptionDefinitionId (p. 452)**

The ID of the subscription definition.

- where used: path; required: true
- type: string

**SubscriptionDefinitionVersionId (p. 452)**

The ID of the subscription definition version.

- where used: path; required: true
- type: string

**Responses:**

**200 (GetSubscriptionDefinitionVersionResponse)**

```
GetSubscriptionDefinitionVersionResponse (p. 492)
{
"Arn": "string",
"Id": "string",
"Version": "string",
"CreationTimestamp": "string",
"Definition": {
"Subscriptions": [
{
"Id": "string",
"Source": "string",
"Subject": "string",
"Target": "string"
}
]
}
```

GetSubscriptionDefinitionVersionResponse

Information about a subscription definition version.

- type: object

**Arn**

The ARN of the subscription definition version.

- type: string

**Id**

The ID of the subscription definition version.
type: string
Version
The version of the subscription definition version.
type: string
CreationTimestamp
The time, in milliseconds since the epoch, when the subscription definition version was created.
type: string
Definition
Information about a subscription definition version.
type: object
Subscriptions
A list of subscriptions.
type: array
items: Subscription (p. 518)
Subscription
Information about a subscription.
type: object
required: ["Id", "Source", "Subject", "Target"]
Id
The id of the subscription.
type: string
Source
The source of the subscription. Can be a thing ARN, a Lambda function ARN, 'cloud' (which represents the AWS IoT cloud), or 'GGShadowService'.
type: string
Subject
The subject of the message.
type: string
Target
Where the message is sent to. Can be a thing ARN, a Lambda function ARN, 'cloud' (which represents the AWS IoT cloud), or 'GGShadowService'.
type: string

400
invalid request

GeneralError (p. 476)
"Message": "string",
"ErrorDetails": [
  {
    "DetailedErrorCode": "string",
    "DetailedErrorMessage": "string"
  }
]
}

**GeneralError**

General error information.

- **type**: object
- **required**: ["Message"]

**Message**

A message containing information about the error.

- **type**: string

**ErrorDetails**

A list of error details.

- **type**: array
- **items**: [ErrorDetail](p. 468)

**ErrorDetail**

Details about the error.

- **type**: object

- **DetailedErrorCode**
  
  - **type**: string

- **DetailedErrorMessage**
  
  - **type**: string

---

**ListCoreDefinitionVersions**

Lists the versions of a core definition.

**URI:** GET /greengrass/definition/cores/CoreDefinitionId/versions

**Produces:** application/json

**CLI:**

```
aws greengrass list-core-definition-versions \
  --core-definition-id <value> \n  [--next-token <value>] \n  [--max-results <value>] \
```
AWS Greengrass API Reference
ListCoreDefinitionVersions

[--cli-input-json <value>]
[--generate-cli-skeleton]

cli-input-json format:

{
   "CoreDefinitionId": "string",
   "NextToken": "string",
   "MaxResults": "integer"
}

Parameters:

CoreDefinitionId (p. 430)

The ID of the core definition.

where used: path; required: true
type: string

NextToken (p. 451)

The token for the next set of results, or null if there are no additional results.

where used: query; required: false
type: string

MaxResults (p. 450)

The maximum number of results to be returned per request.

where used: query; required: false
type: integer

Responses:

200 (ListCoreDefinitionVersionsResponse)

ListVersionsResponse (p. 502)

{
   "Versions": [
      {
         "Arn": "string",
         "Id": "string",
         "Version": "string",
         "CreationTimestamp": "string"
      },
      "NextToken": "string"
   ]
}

ListVersionsResponse

A list of versions.
type: object
Versions
  Information about a version.
  type: array
  items: VersionInformation (p. 521)

VersionInformation
  Information about a version.
  type: object

Arn
  The ARN of the version.
  type: string

Id
  The ID of the version.
  type: string

Version
  The unique ID of the version.
  type: string

CreationTimestamp
  The time, in milliseconds since the epoch, when the version was created.
  type: string

NextToken
  The token for the next set of results, or null if there are no additional results.
  type: string

400
  invalid request

GeneralError (p. 476)

  {  
    "Message": "string",
    "ErrorDetails": [  
      {  
        "DetailedErrorCode": "string",
        "DetailedErrorMessage": "string"
      }  
    ]
  }

GeneralError
  General error information.
  type: object
  required: ["Message"]
ListCoreDefinitions

Retrieves a list of core definitions.

URI: GET /greengrass/definition/cores

Produces: application/json

CLI:

aws greengrass list-core-definitions \  
   [--max-results <value>] \  
   [--next-token <value>] \  
   [--cli-input-json <value>] \  
   [--generate-cli-skeleton]

cli-input-json format:

{
   "MaxResults": "integer",
   "NextToken": "string"
}

Parameters:

MaxResults (p. 450)

The maximum number of results to be returned per request.
where used: query; required: false
type: integer

NextToken

The token for the next set of results, or null if there are no additional results.
where used: query; required: false
type: string

Responses:

200 (ListCoreDefinitionsResponse)

ListDefinitionsResponse

A list of definitions.
type: object

Definitions

Information about a definition.
type: array

items: DefinitionInformation

DefinitionInformation

Information about a definition.
type: object

Name

The name of the definition.
type: string

Id

The ID of the definition.
type: string
Arn
   The ARN of the definition.
   type: string
LastUpdatedAtTimestamp
   The time, in milliseconds since the epoch, when the definition was last updated.
   type: string
CreationTimestamp
   The time, in milliseconds since the epoch, when the definition was created.
   type: string
LatestVersion
   The latest version of the definition.
   type: string
LatestVersionArn
   The ARN of the latest version of the definition.
   type: string
NextToken
   The token for the next set of results, or null if there are no additional results.
   type: string

**ListDeployments**

Returns a history of deployments for the group.

**URI:** GET /greengrass/groups/GroupID/deployments

**Produces:** application/json

**CLI:**

```bash
aws greengrass list-deployments \
   --group-id <value> \ 
   [--max-results <value>] \ 
   [--next-token <value>] \ 
   [--cli-input-json <value>] \ 
   [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
   "GroupId": "string",
   "MaxResults": "integer",
   "NextToken": "string"
}
```
Parameters:

GroupId (p. 449)
The ID of the AWS Greengrass group.
where used: path; required: true
type: string

MaxResults (p. 450)
The maximum number of results to be returned per request.
where used: query; required: false
type: integer

NextToken (p. 451)
The token for the next set of results, or null if there are no additional results.
where used: query; required: false
type: string

Responses:

200
Success. The response body contains the list of deployments for the given group.

ListDeploymentsResponse (p. 499)

```
{
  "Deployments": [
    {
      "GroupArn": "string",
      "DeploymentId": "string",
      "DeploymentArn": "string",
      "DeploymentType": "NewDeployment|Redeployment|ResetDeployment|ForceResetDeployment",
      "CreatedAt": "string"
    }
  ],
  "NextToken": "string"
}
```

ListDeploymentsResponse
type: object

Deployments
type: array

items: Deployment (p. 464)

Deployment
Information about a deployment.
type: object
**GroupArn**

The ARN of the group for this deployment.

*type: string*

**DeploymentId**

The ID of the deployment.

*type: string*

**DeploymentArn**

The ARN of the deployment.

*type: string*

**DeploymentType**

*type: string*
*enum: ["NewDeployment", "Redeployment", "ResetDeployment", "ForceResetDeployment"]*

**CreatedAt**

The time, in milliseconds since the epoch, when the deployment was created.

*type: string*

**NextToken**

The token for the next set of results, or null if there are no additional results.

*in: query*
*type: string*

**400**

invalid request

**GeneralError** *(p. 476)*

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

**GeneralError**

General error information.

*type: object*

**required: ["Message"]*

**Message**

A message containing information about the error.

*type: string*
ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

ListDeviceDefinitionVersions

Lists the versions of a device definition.

URI: GET /greengrass/definition/devices/{DeviceDefinitionId}/versions

Produces: application/json

CLI:

```
aws greengrass list-device-definition-versions
   --device-definition-id <value> \ 
   [--next-token <value>] \ 
   [--max-results <value>] \ 
   [--cli-input-json <value>] \ 
   [--generate-cli-skeleton]
```

cli-input-json format:

```
{
   "DeviceDefinitionId": "string",
   "NextToken": "string",
   "MaxResults": "integer"
}
```

Parameters:

DeviceDefinitionId (p. 448)

The ID of the device definition.

where used: path; required: true

type: string
**NextToken (p. 451)**

The token for the next set of results, or `null` if there are no additional results.

where used: query; required: false

type: string

**MaxResults (p. 450)**

The maximum number of results to be returned per request.

where used: query; required: false

type: integer

**Responses:**

**200** *(ListDeviceDefinitionVersionsResponse)*

**ListVersionsResponse (p. 502)**

```json
{
   "Versions": [
      {
         "Arn": "string",
         "Id": "string",
         "Version": "string",
         "CreationTimestamp": "string"
      }
   ],
   "NextToken": "string"
}
```

ListVersionsResponse

A list of versions.

type: object

Versions

Information about a version.

type: array

items: **VersionInformation (p. 521)**

VersionInformation

Information about a version.

type: object

Arn

The ARN of the version.

type: string

Id

The ID of the version.
ListDeviceDefinitionVersions

- `type: string`
  - `Version`
    - The unique ID of the version.
  - `CreationTimestamp`
    - The time, in milliseconds since the epoch, when the version was created.
  - `NextToken`
    - The token for the next set of results, or `null` if there are no additional results.

400

**invalid request**

**GeneralError (p. 476)**

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

**GeneralError**

- General error information.
  - `type: object`
    - required: `"Message"`

**Message**

- A message containing information about the error.
  - `type: string`

**ErrorDetails**

- A list of error details.
  - `type: array`
    - `items: ErrorDetail (p. 468)`

**ErrorDetail**

- Details about the error.
  - `type: object`

**DetailedErrorCode**

- A detailed error code.
ListDeviceDefinitions

Retrieves a list of device definitions.

**URI:** `GET /greengrass/definition/devices`

**Produces:** `application/json`

**CLI:**

```bash
cli-input-json format:
{
   "MaxResults": "integer",
   "NextToken": "string"
}
```

**Parameters:**

**MaxResults** *(p. 450)*

The maximum number of results to be returned per request.

- where used: query; required: false

  type: integer

**NextToken** *(p. 451)*

The token for the next set of results, or `null` if there are no additional results.

- where used: query; required: false

  type: string

**Responses:**

**200** *(ListDeviceDefinitionsResponse)*

ListDefinitionsResponse *(p. 498)*

```json
{
}
```
"Definitions": [ 
{
   "Name": "string",
   "Id": "string",
   "Arn": "string",
   "LastUpdatedTimestamp": "string",
   "CreationTimestamp": "string",
   "LatestVersion": "string",
   "LatestVersionArn": "string"
}
],
"NextToken": "string"
}

ListDefinitionsResponse

A list of definitions.

type: object

Definitions

Information about a definition.

type: array

items: DefinitionInformation (p. 463)

DefinitionInformation

Information about a definition.

type: object

Name

The name of the definition.

type: string

Id

The ID of the definition.

type: string

Arn

The ARN of the definition.

type: string

LastUpdatedTimestamp

The time, in milliseconds since the epoch, when the definition was last updated.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the definition was created.

type: string

LatestVersion

The latest version of the definition.
ListFunctionDefinitionVersions

Lists the versions of a Lambda function definition.

URL: GET /greengrass/definition/functions/FunctionDefinitionId/versions

Produces: application/json

CLI:

```
aws greengrass list-function-definition-versions
  --function-definition-id <value> \n  [--next-token <value>] \n  [--max-results <value>] \n  [--cli-input-json <value>] \n  [--generate-cli-skeleton]
```

cli-input-json format:

```
{
  "FunctionDefinitionId": "string",
  "NextToken": "string",
  "MaxResults": "integer"
}
```

Parameters:

FunctionDefinitionId (p. 449)

The ID of the Lambda function definition.

where used: path; required: true

type: string

NextToken (p. 451)

The token for the next set of results, or null if there are no additional results.

where used: query; required: false

type: string

MaxResults (p. 450)

The maximum number of results to be returned per request.
where used: query; required: false

type: integer

**Responses:**

**200** (ListFunctionDefinitionVersionsResponse)

ListFunctionDefinitionVersionsResponse (p. 502)

```json
{
  "Versions": [
    {
      "Arn": "string",
      "Id": "string",
      "Version": "string",
      "CreationTimestamp": "string"
    }
  ],
  "NextToken": "string"
}
```

ListFunctionDefinitionVersionsResponse

A list of versions.

type: object

Versions

Information about a version.

type: array

items: VersionInformation (p. 521)

VersionInformation

Information about a version.

type: object

Arn

The ARN of the version.

type: string

Id

The ID of the version.

type: string

Version

The unique ID of the version.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the version was created.

type: string
NextToken

The token for the next set of results, or `null` if there are no additional results.

type: string

400

invalid request

**GeneralError (p. 476)**

```json
{
   "Message": "string",
   "ErrorDetails": [
      {
         "DetailedErrorCode": "string",
         "DetailedErrorMessage": "string"
      }
   ]
}
```

**GeneralError**

General error information.

type: object

required: ["Message"]

**Message**

A message containing information about the error.

type: string

**ErrorDetails**

A list of error details.

type: array

items: **ErrorDetail (p. 468)**

**ErrorDetail**

Details about the error.

type: object

**DetailedErrorCode**

A detailed error code.

type: string

**DetailedErrorMessage**

A detailed error message.

type: string

---

**ListFunctionDefinitions**

Retrieves a list of Lambda function definitions.
URI: GET /greengrass/definition/functions

Produces: application/json

**CLI:**

```
aws greengrass list-function-definitions
    [--max-results <value>] \
    [--next-token <value>] \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

cli-input-json format:

```
{
    "MaxResults": "integer",
    "NextToken": "string"
}
```

**Parameters:**

**MaxResults (p. 450)**

The maximum number of results to be returned per request.

where used: query; required: false

type: integer

**NextToken (p. 451)**

The token for the next set of results, or null if there are no additional results.

where used: query; required: false

type: string

**Responses:**

**200 (ListFunctionDefinitionsResponse)**

Success. The response contains the IDs of all the Greengrass Lambda function definitions in this account.

**ListDefinitionsResponse (p. 498)**

```
{
    "Definitions": [
        {
            "Name": "string",
            "Id": "string",
            "Arn": "string",
            "LastUpdatedTimestamp": "string",
            "CreationTimestamp": "string",
            "LatestVersion": "string",
            "LatestVersionArn": "string"
        }
    ]
}
```
"NextToken": "string"
}

ListDefinitionsResponse
   A list of definitions.
   type: object
Definitions
   Information about a definition.
   type: array
   items: DefinitionInformation
DefinitionInformation
   Information about a definition.
   type: object
Name
   The name of the definition.
   type: string
Id
   The ID of the definition.
   type: string
Arn
   The ARN of the definition.
   type: string
LastUpdatedTimestamp
   The time, in milliseconds since the epoch, when the definition was last updated.
   type: string
CreationTimestamp
   The time, in milliseconds since the epoch, when the definition was created.
   type: string
LatestVersion
   The latest version of the definition.
   type: string
LatestVersionArn
   The ARN of the latest version of the definition.
   type: string
NextToken
   The token for the next set of results, or null if there are no additional results.
type: string

ListGroupCertificateAuthorities

Retrieves the current CAs for a group.

URI: GET /greengrass/groups/GroupId/certificateauthorities

Produces: application/json

CLI:

aws greengrass list-group-certificate-authorities \
  --group-id <value> \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]

cli-input-json format:

{
  "GroupId": "string"
}

Parameters:

GroupId (p. 449)

The ID of the AWS Greengrass group.
  where used: path; required: true
  type: string

Responses:

200

Success. The response body contains the PKI Configuration.

ListGroupCertificateAuthoritiesResponse (p. 500)

{
  "GroupCertificateAuthorities": [
    {
      "GroupCertificateAuthorityId": "string",
      "GroupCertificateAuthorityArn": "string"
    }
  ]
}

ListGroupCertificateAuthoritiesResponse

type: object
GroupCertificateAuthorities

A list of certificate authorities associated with the group.
type: array

items: GroupCertificateAuthorityProperties (p. 494)

Group Certificate Authority Properties

Information about a certificate authority for a group.

type: object

GroupCertificateAuthorityId

The ID of the certificate authority for the group.

type: string

GroupCertificateAuthorityArn

The ARN of the certificate authority for the group.

type: string

400

invalid request

GeneralError (p. 476)

{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object
DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

500

server error

GeneralError (p. 476)

```
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string
ListGroupVersions

Lists the versions of a group.

URL: GET /greengrass/groups/GroupId/versions

Produces: application/json

CLI:

```bash
aws greengrass list-group-versions \
  --group-id <value> \ 
  [--next-token <value>] \ 
  [--max-results <value>] \ 
  [--cli-input-json <value>] \ 
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "GroupId": "string",
  "NextToken": "string",
  "MaxResults": "integer"
}
```

Parameters:

**GroupId (p. 449)**

The ID of the AWS Greengrass group.

where used: path; required: true

  type: string

**NextToken (p. 451)**

The token for the next set of results, or null if there are no additional results.

where used: query; required: false

  type: string

**MaxResults (p. 450)**

The maximum number of results to be returned per request.

where used: query; required: false

  type: integer

Responses:

**200 (ListGroupVersionsResponse)**

Success. The response contains the list of versions and metadata for the given group.

**ListVersionsResponse (p. 502)**
ListGroupVersions

A list of versions.

**ListVersionsResponse**

A list of versions.

**type:** object

**Versions**

Information about a version.

**type:** array

**items:** VersionInformation (p. 521)

**VersionInformation**

Information about a version.

**type:** object

**Arn**

The ARN of the version.

**type:** string

**Id**

The ID of the version.

**type:** string

**Version**

The unique ID of the version.

**type:** string

**CreationTimestamp**

The time, in milliseconds since the epoch, when the version was created.

**type:** string

**NextToken**

The token for the next set of results, or null if there are no additional results.

**type:** string

**400**

invalid request

**GeneralError (p. 476)**
ListGroups

Retrieves a list of groups.

URI: GET /greengrass/groups

Produces: application/json

CLI:

```bash
aws greengrass list-groups \
  [--max-results <value>] \
  [--next-token <value>]
```
ListGroups

cli-input-json format:

```json
{
    "MaxResults": "integer",
    "NextToken": "string"
}
```

Parameters:

**MaxResults (p. 450)**

The maximum number of results to be returned per request.

- where used: query; required: false
- type: integer

**NextToken (p. 451)**

The token for the next set of results, or null if there are no additional results.

- where used: query; required: false
- type: string

Responses:

**200 (ListGroupsResponse)**

ListGroupsResponse (p. 501)

```json
{
    "Groups": [,
    {
        "Name": "string",
        "Id": "string",
        "Arn": "string",
        "LastUpdatedTimestamp": "string",
        "CreationTimestamp": "string",
        "LatestVersion": "string",
        "LatestVersionArn": "string"
    },
    "NextToken": "string"
}
```

ListGroupsResponse

- type: object
- Groups
  Information about a group.
  - type: array
  - items: GroupInformation (p. 495)
GroupInformation
   Information about a group.
   type: object
Name
   The name of the group.
   type: string
Id
   The ID of the group.
   type: string
Arn
   The ARN of the group.
   type: string
LastUpdatedTimestamp
   The time, in milliseconds since the epoch, when the group was last updated.
   type: string
CreationTimestamp
   The time, in milliseconds since the epoch, when the group was created.
   type: string
LatestVersion
   The latest version of the group.
   type: string
LatestVersionArn
   The ARN of the latest version of the group.
   type: string
NextToken
   The token for the next set of results, or null if there are no additional results.
   type: string

ListLoggerDefinitionVersions
Lists the versions of a logger definition.
URI: GET /greengrass/definition/loggers/LoggerDefinitionId/versions
Produces: application/json
CLI:
```
aws greengrass list-logger-definition-versions \
   --logger-definition-id <value> \
```
ListLoggerDefinitionVersions

cli-input-json format:

```json
{
  "LoggerDefinitionId": "string",
  "NextToken": "string",
  "MaxResults": "integer"
}
```

Parameters:

**LoggerDefinitionId (p. 450)**

The ID of the logger definition.

- where used: path; required: true
- type: string

**NextToken (p. 451)**

The token for the next set of results, or null if there are no additional results.

- where used: query; required: false
- type: string

**MaxResults (p. 450)**

The maximum number of results to be returned per request.

- where used: query; required: false
- type: integer

Responses:

**200 (ListLoggerDefinitionVersionsResponse)**

ListVersionsResponse (p. 502)

```json
{
  "Versions": [
    {
      "Arn": "string",
      "Id": "string",
      "Version": "string",
      "CreationTimestamp": "string"
    }
  ],
  "NextToken": "string"
}
```

ListVersionsResponse

A list of versions.
type: object

Versions
  Information about a version.
  type: array
  items: VersionInformation (p. 521)

VersionInformation
  Information about a version.
  type: object

Arn
  The ARN of the version.
  type: string

Id
  The ID of the version.
  type: string

Version
  The unique ID of the version.
  type: string

CreationTimestamp
  The time, in milliseconds since the epoch, when the version was created.
  type: string

NextToken
  The token for the next set of results, or null if there are no additional results.
  type: string

400
  invalid request

GeneralError (p. 476)

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

GeneralError
  General error information.
  type: object
required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

ListLoggerDefinitions

Retrieves a list of logger definitions.

URI: GET /greengrass/definition/loggers

Produces: application/json

CLI:

```
aws greengrass list-logger-definitions \
   [--max-results <value>] \
   [--next-token <value>] \
   [--cli-input-json <value>] \
   [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
   "MaxResults": "integer",
   "NextToken": "string"
}
```

Parameters:

MaxResults (p. 450)

The maximum number of results to be returned per request.
where used: query; required: false
type: integer

**NextToken (p. 451)**

The token for the next set of results, or `null` if there are no additional results.

where used: query; required: false
type: string

**Responses:**

**200 (ListLoggerDefinitionsResponse)**

ListDefinitionsResponse (p. 498)

```json
{
  "Definitions": [
    {
      "Name": "string",
      "Id": "string",
      "Arn": "string",
      "LastUpdatedTimestamp": "string",
      "CreationTimestamp": "string",
      "LatestVersion": "string",
      "LatestVersionArn": "string"
    }
  ],
  "NextToken": "string"
}
```

ListDefinitionsResponse

A list of definitions.

type: object

**Definitions**

Information about a definition.

type: array

items: DefinitionInformation (p. 463)

**DefinitionInformation**

Information about a definition.

type: object

**Name**

The name of the definition.

type: string

**Id**

The ID of the definition.

type: string
Arn

The ARN of the definition.

type: string

LastUpdatedTimestamp

The time, in milliseconds since the epoch, when the definition was last updated.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the definition was created.

type: string

LatestVersion

The latest version of the definition.

type: string

LatestVersionArn

The ARN of the latest version of the definition.

type: string

NextToken

The token for the next set of results, or null if there are no additional results.

type: string

ListResourceDefinitionVersions

Lists the versions of a resource definition.

URL: GET /greengrass/definition/resources/<ResourceDefinitionId>/versions

Produces: application/json

CLI:

```
aws greengrass list-resource-definition-versions \\
    --resource-definition-id <value> \\
    [--next-token <value>] \\
    [--max-results <value>] \\
    [--cli-input-json <value>] \\
    [--generate-cli-skeleton]
```

cli-input-json format:

```
{
    "ResourceDefinitionId": "string",
    "NextToken": "string",
    "MaxResults": "integer"
}
```
Parameters:

**ResourceDefinitionId** *(p. 452)*

The ID of the resource definition.

where used: path; required: true
type: string

**NextToken** *(p. 451)*

The token for the next set of results, or null if there are no additional results.

where used: query; required: false
type: string

**MaxResults** *(p. 450)*

The maximum number of results to be returned per request.

where used: query; required: false
type: integer

Responses:

**200** *(ListResourceDefinitionVersionsResponse)*

**ListVersionsResponse** *(p. 502)*

```json
{
  "Versions": [
    {
      "Arn": "string",
      "Id": "string",
      "Version": "string",
      "CreationTimestamp": "string"
    },
    {
      "Arn": "string",
      "Id": "string",
      "Version": "string",
      "CreationTimestamp": "string"
    }
  ],
  "NextToken": "string"
}
```

**ListVersionsResponse**

A list of versions.
type: object

**Versions**

Information about a version.
type: array

items: **VersionInformation** *(p. 521)*

**VersionInformation**

Information about a version.
type: object
Arn

The ARN of the version.

type: string

Id

The ID of the version.

type: string

Version

The unique ID of the version.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the version was created.

type: string

NextToken

The token for the next set of results, or null if there are no additional results.

type: string

400

invalid request

GeneralError (p. 476)

{
    "Message": "string",
    "ErrorDetails": [
        {
            "DetailedErrorCode": "string",
            "DetailedErrorMessage": "string"
        }
    ]
}

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array
items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

ListResourceDefinitions

Retrieves a list of resource definitions.

URI: GET /greengrass/definition/resources

Produces: application/json

CLI:

```
aws greengrass list-resource-definitions \
    [--max-results <value>] \
    [--next-token <value>] \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

cli-input-json format:

```
{
    "MaxResults": "integer",
    "NextToken": "string"
}
```

Parameters:

**MaxResults (p. 450)**

The maximum number of results to be returned per request.

where used: query; required: false

  type: integer

**NextToken (p. 451)**

The token for the next set of results, or null if there are no additional results.

where used: query; required: false

  type: string
Responses:

200 (ListResourceDefinitionsResponse)

The IDs of all the Greengrass resource definitions in this account.

ListDefinitionsResponse (p. 498)

```json
{
   "Definitions": [
   {
      "Name": "string",
      "Id": "string",
      "Arn": "string",
      "LastUpdatedTimestamp": "string",
      "CreationTimestamp": "string",
      "LatestVersion": "string",
      "LatestVersionArn": "string"
   }
   ],
   "NextToken": "string"
}
```

ListDefinitionsResponse

A list of definitions.

- **type**: object

Definitions

Information about a definition.

- **type**: array

items: DefinitionInformation (p. 463)

DefinitionInformation

Information about a definition.

- **type**: object

Name

The name of the definition.

- **type**: string

Id

The ID of the definition.

- **type**: string

Arn

The ARN of the definition.

- **type**: string

LastUpdatedTimestamp

The time, in milliseconds since the epoch, when the definition was last updated.
ListSubscriptionDefinitionVersions

Lists the versions of a subscription definition.

URI: GET /greengrass/definition/subscriptions/{SubscriptionDefinitionId}/versions

Produces: application/json

CLI:

```
aws greengrass list-subscription-definition-versions
    --subscription-definition-id <value> \n    [--next-token <value>] \n    [--max-results <value>] \n    [--cli-input-json <value>] \n    [--generate-cli-skeleton]
```

cli-input-json format:

```
{
    "SubscriptionDefinitionId": "string",
    "NextToken": "string",
    "MaxResults": "integer"
}
```

Parameters:

**SubscriptionDefinitionId** *(p. 452)*

The ID of the subscription definition.

where used: path; required: true

  type: string
**NextToken (p. 451)**

The token for the next set of results, or `null` if there are no additional results.

where used: query; required: false
type: string

**MaxResults (p. 450)**

The maximum number of results to be returned per request.

where used: query; required: false
type: integer

**Responses:**

**200** (ListSubscriptionDefinitionVersionsResponse)

**ListVersionsResponse (p. 502)**

```
{
   "Versions": [
      {
         "Arn": "string",
         "Id": "string",
         "Version": "string",
         "CreationTimestamp": "string"
      }
   ],
   "NextToken": "string"
}
```

**ListVersionsResponse**

A list of versions.
type: object

**Versions**

Information about a version.
type: array

**items: VersionInformation (p. 521)**

**VersionInformation**

Information about a version.
type: object

**Arn**

The ARN of the version.
type: string

**Id**

The ID of the version.
type: string
Version
The unique ID of the version.
type: string
CreationTimestamp
The time, in milliseconds since the epoch, when the version was created.
type: string
NextToken
The token for the next set of results, or null if there are no additional results.
type: string

400
invalid request

**GeneralError (p. 476)**

```json
{
    "Message": "string",
    "ErrorDetails": [
        {
            "DetailedErrorCode": "string",
            "DetailedErrorMessage": "string"
        }
    ]
}
```

**GeneralError**
General error information.
type: object
required: ["Message"]

**Message**
A message containing information about the error.
type: string

**ErrorDetails**
A list of error details.
type: array
type: object

**ErrorDetail**
Details about the error.
type: object

**DetailedErrorCode**
A detailed error code.
ListSubscriptionDefinitions

Retrieves a list of subscription definitions.

URL: GET /greengrass/definition/subscriptions

Produces: application/json

CLI:

```bash
aws greengrass list-subscription-definitions \\   [--max-results <value>] \\   [--next-token <value>] \\   [--cli-input-json <value>] \\   [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
   "MaxResults": "integer",
   "NextToken": "string"
}
```

Parameters:

**MaxResults** (p. 450)

The maximum number of results to be returned per request.

where used: query; required: false

  type: integer

**NextToken** (p. 451)

The token for the next set of results, or null if there are no additional results.

where used: query; required: false

  type: string

Responses:

**200** (ListSubscriptionDefinitionsResponse)

```json
ListDefinitionsResponse (p. 498)
{
   "Definitions": [
```

```json
```
ListSubscriptionDefinitions

```
{
  "Name": "string",
  "Id": "string",
  "Arn": "string",
  "LastUpdatedTimestamp": "string",
  "CreationTimestamp": "string",
  "LatestVersion": "string",
  "LatestVersionArn": "string"
}
}
"NextToken": "string"
}
```

ListDefinitionsResponse

A list of definitions.

type: object

Definitions

Information about a definition.

type: array

items: DefinitionInformation (p. 463)

DefinitionInformation

Information about a definition.

type: object

Name

The name of the definition.

type: string

Id

The ID of the definition.

type: string

Arn

The ARN of the definition.

type: string

LastUpdatedTimestamp

The time, in milliseconds since the epoch, when the definition was last updated.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the definition was created.

type: string

LatestVersion

The latest version of the definition.

type: string
LatestVersionArn

The ARN of the latest version of the definition.

type: string

NextToken

The token for the next set of results, or null if there are no additional results.

type: string

**ResetDeployments**

Resets a group's deployments.

URI: POST /greengrass/groups/**GroupId**/deployments/$reset

Produces: application/json

**CLI:**

```bash
aws greengrass reset-deployments \
  --group-id <value> \
  [--force | --no-force] \
  [--amzn-client-token <value>] \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "GroupId": "string",
  "Force": "boolean",
  "AmznClientToken": "string"
}
```

**Parameters:**

**GroupId** *(p. 449)*

The ID of the AWS Greengrass group.

where used: path; required: true

type: string

**ResetDeploymentsRequestBody** *(p. 451)*

Information needed to reset deployments.

where used: body; required: true

```json
{
  "Force": true
}
```

schema:
ResetDeploymentsRequest

Information about a group reset request.

type: object

Force

If true, performs a best-effort only core reset.

type: boolean

X-Amzn-Client-Token (p. 455)

A client token used to correlate requests and responses.

where used: header; required: false

type: string

Responses:

200

Success. The group’s deployments were reset.

ResetDeploymentsResponse (p. 508)

```
{
  "DeploymentArn": "string",
  "DeploymentId": "string"
}
```

ResetDeploymentsResponse

type: object

DeploymentArn

The ARN of the deployment.

type: string

DeploymentId

The ID of the deployment.

type: string

400

invalid request

GeneralError (p. 476)

```
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

AWS Greengrass API Reference
UpdateConnectivityInfo

GeneralError

General error information.
type: object
required: ["Message"]
Message

A message containing information about the error.
type: string
ErrorDetails

A list of error details.
type: array
items: ErrorDetail (p. 468)
ErrorDetail

Details about the error.
type: object
DetailedErrorCode

A detailed error code.
type: string
DetailedErrorMessage

A detailed error message.
type: string

UpdateConnectivityInfo

Updates the connectivity information for the core. Any devices that belong to the group which has this core will receive this information in order to find the location of the core and connect to it.

URI: PUT /greengrass/things/<ThingName>/connectivityInfo

Produces: application/json

CLI:

```
aws greengrass update-connectivity-info \
  --thing-name <value> \ 
  [--connectivity-info <value>] \ 
  [--cli-input-json <value>] \ 
  [--generate-cli-skeleton]
```

cli-input-json format:

```
{
  "ThingName": "string",
  "ConnectivityInfo": [
```
AWS Greengrass API Reference
UpdateConnectivityInfo

```json
{
   "Id": "string",
   "HostAddress": "string",
   "PortNumber": 0x01,
   "Metadata": "string"
}
}
```

**Parameters:**

**ThingName (p. 453)**

The thing name.

where used: path; required: true
type: string

**UpdateConnectivityInfoRequestBody (p. 453)**

The connectivity information.

where used: body; required:

```json
{
   "ConnectivityInfo": [
   {
      "Id": "string",
      "HostAddress": "string",
      "PortNumber": 0x01,
      "Metadata": "string"
   }
   ]
}
```

**schema:**

Information required to update a Greengrass core's connectivity.

**ConnectivityInfo**

A list of connectivity info.
type: array
items: ConnectivityInfo (p. 458)

Information about a Greengrass core's connectivity.
required: ["Id", "HostAddress"]

**Id**

The ID of the connectivity information.
type: string

**HostAddress**

The endpoint for the Greengrass core. Can be an IP address or DNS.
type: string
PortNumber

The port of the Greengrass core. Usually 8883.

type: integer
format: int32

Metadata

Metadata for this endpoint.

type: string

Responses:

200

success

UpdateConnectivityInfoResponse (p. 520)

{
"message": "string",
"Version": "string"
}

message

A message about the connectivity info update request.

type: string

Version

The new version of the connectivity info.

type: string

400

invalid request

GeneralError (p. 476)

{
"Message": "string",
"ErrorDetails": [
{
"DetailedErrorCode": "string",
"DetailedErrorMessage": "string"
}
]
}

GeneralError

General error information.

type: object

required: ["Message"]
Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

500

server error

GeneralError (p. 476)

```
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

GeneralError

General error information.


type: object

required: ["Message"]

Message

A message containing information about the error.


type: string

ErrorDetails

A list of error details.


type: array
UpdateCoreDefinition

Updates a core definition.

URI: PUT /greengrass/definition/cores/CoreDefinitionId

Produces: application/json

CLI:

```
aws greengrass update-core-definition \  
   --core-definition-id <value> \  
   [--name <value>] \  
   [--cli-input-json <value>] \  
   [--generate-cli-skeleton]
```

cli-input-json format:

```
{
   "CoreDefinitionId": "string",
   "Name": "string"
}
```

Parameters:

CoreDefinitionId (p. 430)

The ID of the core definition.

where used: path; required: true

type: string

UpdateDefinitionRequestBody (p. 454)

where used: body; required: true

```
{
   "Name": "string"
}
```

ErrorDetail (p. 468)

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string
schema:
Name
    The name of the definition.
    required: true
    type: string

Responses:

200
success
Empty (p. 467)
{
}
Empty Schema
    Empty
type: object

400
invalid request
GeneralError (p. 476)
{
    "Message": "string",
    "ErrorDetails": [
        {
            "DetailedErrorCode": "string",
            "DetailedErrorMessage": "string"
        }
    ]
}
GeneralError
    General error information.
type: object
    required: ["Message"]
Message
    A message containing information about the error.
type: string
ErrorDetails
    A list of error details.
type: array
items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.
type: object
DetailedErrorCode

A detailed error code.
type: string
DetailedErrorMessage

A detailed error message.
type: string

UpdateDeviceDefinition

Updates a device definition.

URI: PUT /greengrass/definition/devices/DeviceDefinitionId

Produces: application/json

CLI:

```bash
aws greengrass update-device-definition \
   --device-definition-id <value> \
   [--name <value>] \ 
   [--cli-input-json <value>] \ 
   [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
   "DeviceDefinitionId": "string",
   "Name": "string"
}
```

Parameters:

DeviceDefinitionId (p. 448)

The ID of the device definition.

where used: path; required: true
type: string

UpdateDefinitionRequestBody (p. 454)

where used: body; required: true

```json
{
   "Name": "string"
}
```
schema:
Name
   The name of the definition.
   required: true
   type: string

Responses:

200
   success
       Empty (p. 467)

   
   
   Empty Schema
       Empty
       type: object

400
   invalid request
       GeneralError (p. 476)

       
       
       GeneralError
           General error information.
           type: object
           required: ["Message"]
       Message
           A message containing information about the error.
           type: string
       ErrorDetails
           A list of error details.
           type: array
items: ErrorDetail (p. 468)
ErrorDetail
Details about the error.
type: object
DetailedErrorCode
A detailed error code.
type: string
DetailedErrorMessage
A detailed error message.
type: string

UpdateFunctionDefinition

Updates a Lambda function definition.

URI: PUT /greengrass/definition/functions/FunctionDefinitionId

Produces: application/json

CLI:

```bash
aws greengrass update-function-definition
  --function-definition-id <value> \n  [--name <value>] \n  [--cli-input-json <value>] \n  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "FunctionDefinitionId": "string",
  "Name": "string"
}
```

Parameters:

FunctionDefinitionId (p. 449)
The ID of the Lambda function definition.
where used: path; required: true
type: string

UpdateDefinitionRequestBody (p. 454)
where used: body; required: true

```json
{
  "Name": "string"
}
```
schema:
Name
   The name of the definition.
   required: true
   type: string

Responses:

200
success
Empty (p. 467)

400
invalid request
GeneralError (p. 476)

GeneralError
   General error information.
   type: object
   required: ["Message"]
Message
   A message containing information about the error.
   type: string
ErrorDetails
   A list of error details.
   type: array
items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

**UpdateGroup**

Updates a group.

**URI:** PUT /greengrass/groups/GroupId

**Produces:** application/json

**CLI:**

```bash
aws greengrass update-group \
  --group-id <value> \
  [--name <value>] \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "GroupId": "string",
  "Name": "string"
}
```

**Parameters:**

**GroupId (p. 449)**

The ID of the AWS Greengrass group.

where used: path; required: true

type: string

**UpdateDefinitionRequestBody (p. 454)**

where used: body; required: true

```json
{
  "Name": "string"
}
```
schema:
Name
  The name of the definition.
  required: true
  type: string

Responses:

200
success

Empty (p. 467)

{ }

Empty Schema

Empty
  type: object

400
invalid request

GeneralError (p. 476)

{ "Message": "string",
  "ErrorDetails": [ 
    { 
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    } 
  ]
}

GeneralError

  General error information.
  type: object
  required: ["Message"]

Message

  A message containing information about the error.
  type: string

ErrorDetails

  A list of error details.
type: array
items: ErrorDetail (p. 468)

ErrorDetail
Details about the error.
type: object
DetailedErrorCode
A detailed error code.
type: string
DetailedErrorMessage
A detailed error message.
type: string

UpdateGroupCertificateConfiguration

Updates the Certificate expiry time for a group.

URI: PUT /greengrass/groups/GroupId/certificateauthorities/configuration/expiry

Produces: application/json

CLI:

```bash
aws greengrass update-group-certificate-configuration \
  --group-id <value> \
  [--certificate-expiry-in-milliseconds <value>] \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "GroupId": "string",
  "CertificateExpiryInMilliseconds": "string"
}
```

Parameters:

GroupId (p. 449)
The ID of the AWS Greengrass group.
where used: path; required: true
type: string

UpdateGroupCertificateConfigurationRequestBody (p. 454)
where used: body; required: true

```json
{
}
```
"CertificateExpiryInMilliseconds": "string"
}

schema:
updateGroupCertificateConfigurationRequest
type: object
required: ["CertificateExpiryInMilliseconds"]
CertificateExpiryInMilliseconds
The amount of time remaining before the certificate expires, in milliseconds.
type: string

Responses:
200
Success. The response body contains the PKI Configuration.

GroupCertificateConfiguration (p. 495)
{
  "GroupId": "string",
  "CertificateAuthorityExpiryInMilliseconds": "string",
  "CertificateExpiryInMilliseconds": "string"
}

GroupCertificateConfiguration
Information about a group certificate configuration.
type: object

GroupId
The ID of the group certificate configuration.
type: string

CertificateAuthorityExpiryInMilliseconds
The amount of time remaining before the certificate authority expires, in milliseconds.
type: string

CertificateExpiryInMilliseconds
The amount of time remaining before the certificate expires, in milliseconds.
type: string

400
invalid request

GeneralError (p. 476)
{

"Message": "string",
"ErrorDetails": [
  {
    "DetailedErrorCode": "string",
    "DetailedErrorMessage": "string"
  }
]
}

GeneralError

  General error information.
  type: object
  required: ["Message"]

Message

  A message containing information about the error.
  type: string

ErrorDetails

  A list of error details.
  type: array
  items: ErrorDetail (p. 468)

ErrorDetail

  Details about the error.
  type: object

DetailedErrorCode

  A detailed error code.
  type: string

DetailedErrorMessage

  A detailed error message.
  type: string

500

  server error

GeneralError (p. 476)
GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

UpdateLoggerDefinition

Updates a logger definition.

URL: PUT /greengrass/definition/loggers/LoggerDefinitionId

Produces: application/json

CLI:

```
aws greengrass update-logger-definition \
   --logger-definition-id <value> \
   [--name <value>] \
   [--cli-input-json <value>] \
   [--generate-cli-skeleton]
```

cli-input-json format:

```
{
   "LoggerDefinitionId": "string",
   "Name": "string"
}
```
Parameters:

**LoggerDefinitionId (p. 450)**

The ID of the logger definition.

where used: path; required: true

type: string

**UpdateDefinitionRequestBody (p. 454)**

where used: body; required: true

```json
{
  "Name": "string"
}
```

schema:
Name

The name of the definition.

required: true

type: string

Responses:

200

success

Empty (p. 467)

```json
{}
```

Empty Schema

Empty

type: object

400

invalid request

GeneralError (p. 476)

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```
GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

UpdateResourceDefinition

Updates a resource definition.

URI: PUT /greengrass/definition/resources/ResourceDefinitionId

Produces: application/json

CLI:

```
aws greengrass update-resource-definition \ 
  --resource-definition-id <value> \ 
  [--name <value>] \ 
  [--cli-input-json <value>] \ 
  [--generate-cli-skeleton]
```

cli-input-json format:
Parameters:

**ResourceDefinitionId (p. 452)**

The ID of the resource definition.

where used: path; required: true
type: string

**UpdateDefinitionRequestBody (p. 454)**

where used: body; required: true

```
{
    "Name": "string"
}
```

<table>
<thead>
<tr>
<th>schema</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
</tr>
<tr>
<td>The name of the definition.</td>
</tr>
<tr>
<td>required: true</td>
</tr>
<tr>
<td>type: string</td>
</tr>
</tbody>
</table>

Responses:

**200**

success

Empty (p. 467)

```
{
}
```

<table>
<thead>
<tr>
<th>Empty Schema</th>
</tr>
</thead>
<tbody>
<tr>
<td>Empty</td>
</tr>
<tr>
<td>type: object</td>
</tr>
</tbody>
</table>

**400**

invalid request

GeneralError (p. 476)

```
{
    "Message": "string",
}
```
"ErrorDetails": [  
  {  
    "DetailedErrorCode": "string",  
    "DetailedErrorMessage": "string"  
  }  
]  
}

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

UpdateSubscriptionDefinition

Updates a subscription definition.

URI: PUT /greengrass/definition/subscriptions/SubscriptioanDefinitionId

Produces: application/json

CLI:

```
aws greengrass update-subscription-definition \  
  --subscription-definition-id <value> \  
  [--name <value>] \  
  [--cli-input-json <value>] 
```
cli-input-json format:

```json
{
    "SubscriptionDefinitionId": "string",
    "Name": "string"
}
```

**Parameters:**

**SubscriptionDefinitionId** *(p. 452)*

The ID of the subscription definition.

- where used: path; required: true
- type: string

**UpdateDefinitionRequestBody** *(p. 454)*

- where used: body; required: true

```json
{
    "Name": "string"
}
```

schema:

**Name**

The name of the definition.

- required: true
- type: string

**Responses:**

**200**

success

- **Empty** *(p. 467)*

  ```json
  {
  
  }
  ```

Empty Schema

- Empty
  - type: object

**400**

invalid request

- **GeneralError** *(p. 476)*
Endpoints

The following endpoints are supported:

- /greengrass/definition/cores (p. 213)
- /greengrass/definition/cores/CoreDefinitionId (p. 219)
- /greengrass/definition/cores/CoreDefinitionId/versions (p. 225)
- /greengrass/definition/cores/CoreDefinitionId/versions/CoreDefinitionVersionId (p. 232)
- /greengrass/definition/devices (p. 235)
- /greengrass/definition/devices/DeviceDefinitionId (p. 241)
/greengrass/definition/cores

GET

GET /greengrass/definition/cores

Operation ID: ListCoreDefinitions (p. 152)

Retrieves a list of core definitions.

Produces: application/json
Query Parameters

MaxResults (p. 450)
The maximum number of results to be returned per request.

where used: query; required: false

type: integer

NextToken (p. 451)
The token for the next set of results, or null if there are no additional results.

where used: query; required: false

type: string

CLI

aws greengrass list-core-definitions \  
  [--max-results <value>] \  
  [--next-token <value>] \  
  [--cli-input-json <value>] \  
  [--generate-cli-skeleton]

cli-input-json format:

```json
{
  "MaxResults": "integer",
  "NextToken": "string"
}
```

Responses

200 (ListCoreDefinitionsResponse)

ListDefinitionsResponse (p. 498)

```json
{
  "Definitions": [
    {
      "Name": "string",
      "Id": "string",
      "Arn": "string",
      "LastUpdatedTimestamp": "string",
      "CreationTimestamp": "string",
      "LatestVersion": "string",
      "LatestVersionArn": "string"
    }
  ],
    "NextToken": "string"
}
```

ListDefinitionsResponse

A list of definitions.

type: object
Definitions

Information about a definition.

type: array

items: DefinitionInformation (p. 463)

DefinitionInformation

Information about a definition.

type: object

Name

The name of the definition.

type: string

Id

The ID of the definition.

type: string

Arn

The ARN of the definition.

type: string

LastUpdatedTimestamp

The time, in milliseconds since the epoch, when the definition was last updated.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the definition was created.

type: string

LatestVersion

The latest version of the definition.

type: string

LatestVersionArn

The ARN of the latest version of the definition.

type: string

NextToken

The token for the next set of results, or null if there are no additional results.

type: string

POST

POST /greengrass/definition/cores

Operation ID: CreateCoreDefinition (p. 8)
Creates a core definition. You may provide the initial version of the core definition now or use CreateCoreDefinitionVersion at a later time. AWS Greengrass groups must each contain exactly one AWS Greengrass core.

Produces: application/json

Body Parameters

CreateCoreDefinitionRequestBody (p. 431)

Information needed to create a core definition.

where used: body; required: true

```
{
   "Name": "string",
   "InitialVersion": {
      "Cores": [
         {
            "Id": "string",
            "ThingArn": "string",
            "CertificateArn": "string",
            "SyncShadow": true
         }
      ]
   }
}
```

Name

The name of the core definition.

Type: string

InitialVersion

Information about a core definition version.

Type: object

Cores

A list of cores in the core definition version.

Type: array

Items: Core (p. 458)

Core

Information about a core.

Type: object

Required: ["Id", "ThingArn", "CertificateArn"]

Id

The ID of the core.

Type: string

ThingArn

The ARN of the thing which is the core.
type: string
CertificateArn

The ARN of the certificate associated with the core.

type: string
SyncShadow

If true, the core's local shadow is automatically synced with the cloud.

type: boolean

Header Parameters

**X-Amzn-Client-Token (p. 455)**

A client token used to correlate requests and responses.

where used: header; required: false
type: string

CLI

```bash
aws greengrass create-core-definition
    [--name <value>] \
    [--initial-version <value>] \
    [--amzn-client-token <value>] \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
    "Name": "string",
    "InitialVersion": {
        "Cores": [
            {
                "Id": "string",
                "ThingArn": "string",
                "CertificateArn": "string",
                "SyncShadow": "boolean"
            }
        ],
        "AmznClientToken": "string"
    }
}
```

Responses

**200 (CreateCoreDefinitionResponse)**

DefinitionInformation (p. 463)

```json
{
    "Name": "string",
    "Id": "string",
```
"Arn": "string",
"LastUpdatedTimestamp": "string",
"CreationTimestamp": "string",
"LatestVersion": "string",
"LatestVersionArn": "string"
}

**DefinitionInformation**

Information about a definition.

type: object

**Name**

The name of the definition.

type: string

**Id**

The ID of the definition.

type: string

**Arn**

The ARN of the definition.

type: string

**LastUpdatedTimestamp**

The time, in milliseconds since the epoch, when the definition was last updated.

type: string

**CreationTimestamp**

The time, in milliseconds since the epoch, when the definition was created.

type: string

**LatestVersion**

The latest version of the definition.

type: string

**LatestVersionArn**

The ARN of the latest version of the definition.

type: string

400

invalid request

**GeneralError (p. 476)**

{  
"Message": "string",
"ErrorDetails": [  
{  
"DetailedErrorCode": "string",
}  
]  
}
"DetailedErrorMessage": "string"
}
]
}

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

/greengrass/definition/cores/CoreDefinitionId

GET

GET /greengrass/definition/cores/CoreDefinitionId

Operation ID: GetCoreDefinition (p. 97)

Retrieves information about a core definition version.

Produces: application/json

Path Parameters

CoreDefinitionId (p. 430)

The ID of the core definition.
where used: path; required: true

type: string

CLI

```
aws greengrass get-core-definition \
--core-definition-id <value> \
[--cli-input-json <value>] \
[--generate-cli-skeleton]
```

cli-input-json format:

```
{
  "CoreDefinitionId": "string"
}
```

Responses

200 (GetCoreDefinitionResponse)

DefinitionInformation (p. 463)

```
{
  "Name": "string",
  "Id": "string",
  "Arn": "string",
  "LastUpdatedTimestamp": "string",
  "CreationTimestamp": "string",
  "LatestVersion": "string",
  "LatestVersionArn": "string"
}
```

DefinitionInformation

Information about a definition.

type: object

Name

The name of the definition.

type: string

Id

The ID of the definition.

   type: string

Arn

The ARN of the definition.

   type: string

LastUpdatedTimestamp

The time, in milliseconds since the epoch, when the definition was last updated.

   type: string
CreationTimestamp

The time, in milliseconds since the epoch, when the definition was created.

    type: string

LatestVersion

The latest version of the definition.

    type: string

LatestVersionArn

The ARN of the latest version of the definition.

    type: string

400

invalid request

GeneralError (p. 476)

```
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

GeneralError

General error information.

    type: object

    required: ["Message"]

Message

A message containing information about the error.

    type: string

ErrorDetails

A list of error details.

    type: array

    items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

    type: object

DetailedErrorCode

A detailed error code.
DELETE

DELETE /greengrass/definition/cores/CoreDefinitionId

Operation ID: DeleteCoreDefinition (p. 75)

Deletes a core definition.

Produces: application/json

Path Parameters

CoreDefinitionId (p. 430)

The ID of the core definition.

where used: path; required: true

type: string

CLI

aws greengrass delete-core-definition \
  --core-definition-id <value> \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]

cli-input-json format:

```
{
  "CoreDefinitionId": "string"
}
```

Responses

200

success

Empty (p. 467)

```
{
}
```

Empty Schema

Empty


type: object
400

invalid request

GeneralError (p. 476)

```
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

GeneralError

General error information.

- type: object
  - required: ["Message"]

Message

A message containing information about the error.

- type: string

ErrorDetails

A list of error details.

- type: array
  - items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

- type: object

DetailedErrorCode

A detailed error code.

- type: string

DetailedErrorMessage

A detailed error message.

- type: string

PUT

PUT /greengrass/definition/cores/CoreDefinitionId

Operation ID: UpdateCoreDefinition (p. 195)

Updates a core definition.

Produces: application/json
Body Parameters

**UpdateDefinitionRequestBody** *(p. 454)*

where used: body; required: true

```
{
   "Name": "string"
}
```

**Name**

The name of the definition.

required: true
type: string

Path Parameters

**CoreDefinitionId** *(p. 430)*

The ID of the core definition.

where used: path; required: true
type: string

CLI

```
aws greengrass update-core-definition \
   --core-definition-id <value> \
   [--name <value>] \
   [--cli-input-json <value>] \
   [--generate-cli-skeleton]
```

cli-input-json format:

```
{
   "CoreDefinitionId": "string",
   "Name": "string"
}
```

Responses

**200**

success

**Empty** *(p. 467)*

```
{
}
```

Empty Schema

Empty
### POST /greengrass/definition/cores/CoreDefinitionId/versions

`POST /greengrass/definition/cores/CoreDefinitionId/versions`

**Description**

Provides a list of version IDs defined in the provided core definition.

**Request Parameters**

- **CoreDefinitionId** (required)
  
  Type: string
  
  The core definition ID.

**Response Codes**

- **200**
  
  Success response.

- **400**
  
  invalid request
  
  GeneralError (p. 476)

```json
{
    "Message": "string",
    "ErrorDetails": [
        {
            "DetailedErrorCode": "string",
            "DetailedErrorMessage": "string"
        }
    ]
}
```

**GeneralError**

General error information.

- **Message**
  
  A message containing information about the error.
  
  Type: string

- **ErrorDetails**
  
  A list of error details.
  
  Type: array

**ErrorDetail**

Details about the error.

- **DetailedErrorCode**
  
  A detailed error code.
  
  Type: string

- **DetailedErrorMessage**
  
  A detailed error message.
  
  Type: string
Operation ID: CreateCoreDefinitionVersion (p. 12)

Creates a version of a core definition that has already been defined. AWS Greengrass groups must each contain exactly one AWS Greengrass core.

Produces: application/json

**Body Parameters**

CreateCoreDefinitionVersionRequestBody (p. 431)

where used: body; required: true

```json
{
  "Cores": [
    {
      "Id": "string",
      "ThingArn": "string",
      "CertificateArn": "string",
      "SyncShadow": true
    }
  ]
}
```

CoreDefinitionVersion

Information about a core definition version.

type: object

Cores

A list of cores in the core definition version.

type: array

items: Core (p. 458)

Core

Information about a core.

type: object

required: ["Id", "ThingArn", "CertificateArn"]

Id

The ID of the core.

type: string

ThingArn

The ARN of the thing which is the core.

type: string

CertificateArn

The ARN of the certificate associated with the core.

type: string
SyncShadow

If true, the core's local shadow is automatically synced with the cloud.

type: boolean

Header Parameters

X-Amzn-Client-Token (p. 455)

A client token used to correlate requests and responses.

where used: header; required: false

type: string

Path Parameters

CoreDefinitionId (p. 430)

The ID of the core definition.

where used: path; required: true

type: string

CLI

```bash
aws greengrass create-core-definition-version \
  --core-definition-id <value> \
  [--cores <value>] \
  [--amzn-client-token <value>] \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "CoreDefinitionId": "string",
  "Cores": [
    {
      "Id": "string",
      "ThingArn": "string",
      "CertificateArn": "string",
      "SyncShadow": "boolean"
    }
  ],
  "AmznClientToken": "string"
}
```

Responses

200 (CreateCoreDefinitionVersionResponse)

VersionInformation (p. 521)
"Arn": "string",
"Id": "string",
"Version": "string",
"CreationTimestamp": "string"
}

VersionInformation

Information about a version.
type: object

Arn

The ARN of the version.
type: string

Id

The ID of the version.
type: string

Version

The unique ID of the version.
type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the version was created.
type: string

400

invalid request

GeneralError (p. 476)

{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}

GeneralError

General error information.
type: object

required: ["Message"]

Message

A message containing information about the error.
type: string
ErrorDetails
   A list of error details.
   type: array
   items: ErrorDetail (p. 468)

ErrorDetail
   Details about the error.
   type: object
   DetailedErrorCode
      A detailed error code.
      type: string
   DetailedErrorMessage
      A detailed error message.
      type: string

GET
GET /greengrass/definition/cores/CoreDefinitionId/versions

Operation ID: ListCoreDefinitionVersions (p. 149)

Lists the versions of a core definition.

Produces: application/json

Path Parameters

CoreDefinitionId (p. 430)
   The ID of the core definition.
   where used: path; required: true
   type: string

Query Parameters

NextToken (p. 451)
   The token for the next set of results, or null if there are no additional results.
   where used: query; required: false
   type: string

MaxResults (p. 450)
   The maximum number of results to be returned per request.
   where used: query; required: false
type: integer

CLI

aws greengrass list-core-definition-versions \
   --core-definition-id <value> \
   [--next-token <value>] \
   [--max-results <value>] \
   [--cli-input-json <value>] \
   [--generate-cli-skeleton]

cli-input-json format:

{
   "CoreDefinitionId": "string",
   "NextToken": "string",
   "MaxResults": "integer"
}

Responses

200 (ListCoreDefinitionVersionsResponse)

ListVersionsResponse (p. 502)

{
   "Versions": [
   {
      "Arn": "string",
      "Id": "string",
      "Version": "string",
      "CreationTimestamp": "string"
   }
   ],
   "NextToken": "string"
}

ListVersionsResponse

A list of versions.

  type: object

Versions

  Information about a version.

    type: array

    items: VersionInformation (p. 521)

VersionInformation

  Information about a version.

    type: object

Arn

  The ARN of the version.

    type: string
Id
   The ID of the version.
   type: string
Version
   The unique ID of the version.
   type: string
CreationTimestamp
   The time, in milliseconds since the epoch, when the version was created.
   type: string
NextToken
   The token for the next set of results, or null if there are no additional results.
   type: string

400
invalid request

GeneralError (p. 476)

{
   "Message": "string",
   "ErrorDetails": [
      {
         "DetailedErrorCode": "string",
         "DetailedErrorMessage": "string"
      }
   ]
}

GeneralError
   General error information.
   type: object
   required: ["Message"]
Message
   A message containing information about the error.
   type: string
ErrorDetails
   A list of error details.
   type: array
   items: ErrorDetail (p. 468)
ErrorDetail
   Details about the error.
   type: object
DetailedErrorCode
A detailed error code.
type: string
DetailedErrorMessage
A detailed error message.
type: string

/greengrass/definition/cores/CoreDefinitionId/versions/CoreDefinitionVersionId

GET
GET /greengrass/definition/cores/CoreDefinitionId/versions/CoreDefinitionVersionId

Operation ID: GetCoreDefinitionVersion (p. 100)
Retrieves information about a core definition version.
Produces: application/json

Path Parameters
CoreDefinitionId (p. 430)
The ID of the core definition.
where used: path; required: true
type: string
CoreDefinitionVersionId (p. 430)
The ID of the core definition version.
where used: path; required: true
type: string

CLI
aws greengrass get-core-definition-version
  --core-definition-id <value> \
  --core-definition-version-id <value> \n  [--cli-input-json <value>] \n  [--generate-cli-skeleton]

cli-input-json format:

```json
{
  "CoreDefinitionId": "string",
  "CoreDefinitionVersionId": "string"
}
```
Responses

200

success

GetCoreDefinitionVersionResponse (p. 478)

```json
{
  "Arn": "string",
  "Id": "string",
  "Version": "string",
  "CreationTimestamp": "string",
  "Definition": {
    "Cores": [
      {
        "Id": "string",
        "ThingArn": "string",
        "CertificateArn": "string",
        "SyncShadow": true
      }
    ]
  }
}
```

GetCoreDefinitionVersionResponse

  type: object

Arn

  The ARN of the core definition version.

  type: string

Id

  The ID of the core definition version.

  type: string

Version

  The version of the core definition version.

  type: string

CreationTimestamp

  The time, in milliseconds since the epoch, when the core definition version was created.

  type: string

Definition

  Information about a core definition version.

  type: object

Cores

  A list of cores in the core definition version.

  type: array

  items: Core (p. 458)
Core

Information about a core.
type: object
required: ["Id", "ThingArn", "CertificateArn"]

Id
The ID of the core.
type: string

ThingArn
The ARN of the thing which is the core.
type: string

CertificateArn
The ARN of the certificate associated with the core.
type: string

SyncShadow
If true, the core's local shadow is automatically synced with the cloud.
type: boolean

400
invalid request

GeneralError (p. 476)

{
    "Message": "string",
    "ErrorDetails": [
        {
            "DetailedErrorCode": "string",
            "DetailedErrorMessage": "string"
        }
    ]
}

GeneralError
General error information.
type: object
required: ["Message"]

Message
A message containing information about the error.
type: string

ErrorDetails
A list of error details.
type: array
ErrorDetail (p. 468)
ErrorDetail
Details about the error.
  type: object
DetailedErrorCode
  A detailed error code.
  type: string
DetailedErrorMessage
  A detailed error message.
  type: string

/greengrass/definition/devices

GET
GET /greengrass/definition/devices
Operation ID: ListDeviceDefinitions (p. 160)
Retrieves a list of device definitions.
Produces: application/json

Query Parameters
MaxResults (p. 450)
The maximum number of results to be returned per request.
  where used: query; required: false
  type: integer
NextToken (p. 451)
The token for the next set of results, or null if there are no additional results.
  where used: query; required: false
  type: string

CLI
aws greengrass list-device-definitions \\ [--max-results <value>] \\ [--next-token <value>] \\ [--cli-input-json <value>] \\ [--generate-cli-skeleton]

cli-input-json format:
{

"MaxResults": "integer",
"NextToken": "string"
}

Responses

200 (ListDeviceDefinitionsResponse)

ListDefinitionsResponse (p. 498)

{
  "Definitions": [
    {
      "Name": "string",
      "Id": "string",
      "Arn": "string",
      "LastUpdatedTimestamp": "string",
      "CreationTimestamp": "string",
      "LatestVersion": "string",
      "LatestVersionArn": "string"
    }
  ],
  "NextToken": "string"
}

ListDefinitionsResponse

A list of definitions.

type: object

Definitions

Information about a definition.

type: array

items: DefinitionInformation (p. 463)

DefinitionInformation

Information about a definition.

type: object

Name

The name of the definition.

type: string

Id

The ID of the definition.

type: string

Arn

The ARN of the definition.

type: string

LastUpdatedTimestamp

The time, in milliseconds since the epoch, when the definition was last updated.
CreationTimestamp
The time, in milliseconds since the epoch, when the definition was created.

LatestVersion
The latest version of the definition.

LatestVersionArn
The ARN of the latest version of the definition.

NextToken
The token for the next set of results, or `null` if there are no additional results.

POST
POST /greengrass/definition/devices

Operation ID: CreateDeviceDefinition (p. 18)

Creates a device definition. You may provide the initial version of the device definition now or use CreateDeviceDefinitionVersion at a later time.

Produces: application/json

Body Parameters
CreateDeviceDefinitionRequestBody (p. 433)

Information needed to create a device definition.

where used: body; required: true

```json
{
  "Name": "string",
  "InitialVersion": {
    "Devices": [
      {
        "Id": "string",
        "ThingArn": "string",
        "CertificateArn": "string",
        "SyncShadow": true
      }
    ]
  }
}
```

Name
The name of the device definition.
type: string

InitialVersion

Information about a device definition version.

    type: object

Devices

A list of devices in the definition version.

    type: array

    items: Device (p. 465)

Device

Information about a device.

    type: object

    required: ["Id", "ThingArn", "CertificateArn"]

Id

The ID of the device.

    type: string

ThingArn

The thing ARN of the device.

    type: string

CertificateArn

The ARN of the certificate associated with the device.

    type: string

SyncShadow

If true, the device's local shadow will be automatically synced with the cloud.

    type: boolean

Header Parameters

X-Amzn-Client-Token (p. 455)

A client token used to correlate requests and responses.

    where used: header; required: false

    type: string

CLI

```
aws greengrass create-device-definition \
    [--name <value>] \
    [--initial-version <value>] \
    [--amzn-client-token <value>] \
    [--cli-input-json <value>] \
```
cli-input-json format:

```json
{
    "Name": "string",
    "InitialVersion": {
        "Devices": [
            {
                "Id": "string",
                "ThingArn": "string",
                "CertificateArn": "string",
                "SyncShadow": "boolean"
            }
        ],
        "AmznClientToken": "string"
    }
}
```

Responses

**200** (CreateDeviceDefinitionResponse)

DefinitionInformation (p. 463)

```json
{
    "Name": "string",
    "Id": "string",
    "Arn": "string",
    "LastUpdatedTimestamp": "string",
    "CreationTimestamp": "string",
    "LatestVersion": "string",
    "LatestVersionArn": "string"
}
```

DefinitionInformation

Information about a definition.

- **Name**
  
  The name of the definition.
  
  type: string

- **Id**
  
  The ID of the definition.
  
  type: string

- **Arn**
  
  The ARN of the definition.
  
  type: string

- **LastUpdatedTimestamp**
  
  The time, in milliseconds since the epoch, when the definition was last updated.
  
  type: string

---

239
CreationTimestamp

The time, in milliseconds since the epoch, when the definition was created.

type: string

LatestVersion

The latest version of the definition.

type: string

LatestVersionArn

The ARN of the latest version of the definition.

type: string

400

invalid request

**GeneralError (p. 476)**

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

**GeneralError**

General error information.

type: object

required: ["Message"]

**Message**

A message containing information about the error.

type: string

**ErrorDetails**

A list of error details.

type: array

items: **ErrorDetail (p. 468)**

**ErrorDetail**

Details about the error.

type: object

**DetailedErrorCode**

A detailed error code.

type: string
DetailedErrorMessage

A detailed error message.

type: string

/greengrass/definition/devices/DeviceDefinitionId

GET

GET /greengrass/definition/devices/DeviceDefinitionId

Operation ID: GetDeviceDefinition (p. 105)

Retrieves information about a device definition.

Produces: application/json

Path Parameters

DeviceDefinitionId (p. 448)

The ID of the device definition.

where used: path; required: true

type: string

CLI

aws greengrass get-device-definition \
  --device-definition-id <value> \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]

cli-input-json format:

{
  "DeviceDefinitionId": "string"
}

Responses

200 (GetDeviceDefinitionResponse)

DefinitionInformation (p. 463)

{
  "Name": "string",
  "Id": "string",
  "Arn": "string",
  "LastUpdatedTimestamp": "string",
  "CreationTimestamp": "string",
  "LatestVersion": "string",
  "LatestVersionArn": "string"
}
DefinitionInformation
  Information about a definition.
  type: object
Name
  The name of the definition.
  type: string
Id
  The ID of the definition.
  type: string
Arn
  The ARN of the definition.
  type: string
LastUpdatedTimestamp
  The time, in milliseconds since the epoch, when the definition was last updated.
  type: string
CreationTimestamp
  The time, in milliseconds since the epoch, when the definition was created.
  type: string
LatestVersion
  The latest version of the definition.
  type: string
LatestVersionArn
  The ARN of the latest version of the definition.
  type: string

400
invalid request

**GeneralError (p. 476)**

```json
{
"Message": "string",
"ErrorDetails": [
{
  "DetailedErrorCode": "string",
  "DetailedErrorMessage": "string"
}
]
}
```

GeneralError
General error information.
type: object
required: ["Message"]

Message
A message containing information about the error.
type: string

ErrorDetails
A list of error details.
type: array
items: ErrorDetail (p. 468)

ErrorDetail
Details about the error.
type: object

DetailedErrorCode
A detailed error code.
type: string

DetailedErrorMessage
A detailed error message.
type: string

DELETE
DELETE /greengrass/definition/devices/{DeviceDefinitionId}

Operation ID: DeleteDeviceDefinition (p. 77)

Deletes a device definition.

Produces: application/json

Path Parameters

DeviceDefinitionId (p. 448)
The ID of the device definition.
where used: path; required: true
type: string

CLI

aws greengrass delete-device-definition
   --device-definition-id <value> \
   [--cli-input-json <value>] \
   [--generate-cli-skeleton]
cli-input-json format:

```json
{
  "DeviceDefinitionId": "string"
}
```

Responses

200

success

Empty (p. 467)

```json
{
}
```

Empty Schema

Empty
type: object

400

invalid request

GeneralError (p. 476)

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

GeneralError

General error information.
type: object
required: ["Message"]

Message

A message containing information about the error.
type: string

ErrorDetails

A list of error details.
type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.
type: object
DetailedErrorCode
   A detailed error code.
   type: string
DetailedErrorMessage
   A detailed error message.
   type: string

PUT
PUT /greengrass/definition/devices/DeviceDefinitionId
Operation ID: UpdateDeviceDefinition (p. 197)
Updates a device definition.
Produces: application/json

Body Parameters
UpdateDefinitionRequestBody (p. 454)
where used: body; required: true

```json
{
   "Name": "string"
}
```

Name
   The name of the definition.
   required: true
type: string

Path Parameters
DeviceDefinitionId (p. 448)
The ID of the device definition.
where used: path; required: true
type: string

CLI
aws greengrass update-device-definition \
   --device-definition-id <value> \  
   [--name <value>] \  
   [--cli-input-json <value>] \  
   [--generate-cli-skeleton]
cli-input-json format:

```json
{
   "DeviceDefinitionId": "string",
   "Name": "string"
}
```

**Responses**

**200**

success

Empty (p. 467)

```json
{}
```

Empty Schema

- Empty
  - type: object

**400**

invalid request

GeneralError (p. 476)

```json
{
   "Message": "string",
   "ErrorDetails": [
      {
         "DetailedErrorCode": "string",
         "DetailedErrorMessage": "string"
      }
   ]
}
```

GeneralError

- General error information.
  - type: object
    - required: ["Message"]

Message

- A message containing information about the error.
  - type: string

ErrorDetails

- A list of error details.
  - type: array
    - items: ErrorDetail (p. 468)
ErrorDetail
  Details about the error.
  type: object
DetailedErrorCode
  A detailed error code.
  type: string
DetailedErrorMessage
  A detailed error message.
  type: string

/greengrass/definition/devices/DeviceDefinitionId/versions

POST

POST /greengrass/definition/devices/DeviceDefinitionId/versions

Operation ID: CreateDeviceDefinitionVersion (p. 22)

Creates a version of a device definition that has already been defined.

Produces: application/json

Body Parameters

CreateDeviceDefinitionVersionRequestBody (p. 433)

where used: body; required: true

```
{
  "Devices": [
    {
      "Id": "string",
      "ThingArn": "string",
      "CertificateArn": "string",
      "SyncShadow": true
    }
  ]
}
```

DeviceDefinitionVersion

Information about a device definition version.

  type: object

Devices

A list of devices in the definition version.

  type: array

  items: Device (p. 465)
Device

Information about a device.

type: object

required: ["Id", "ThingArn", "CertificateArn"]

Id

The ID of the device.

type: string

ThingArn

The thing ARN of the device.

type: string

CertificateArn

The ARN of the certificate associated with the device.

type: string

SyncShadow

If true, the device's local shadow will be automatically synced with the cloud.

type: boolean

Header Parameters

X-Amzn-Client-Token (p. 455)

A client token used to correlate requests and responses.

where used: header; required: false

type: string

Path Parameters

DeviceDefinitionId (p. 448)

The ID of the device definition.

where used: path; required: true

type: string

CLI

```
aws greengrass create-device-definition-version \
  --device-definition-id <value> \
  [--devices <value>] \
  [--amzn-client-token <value>] \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:
{
  "DeviceDefinitionId": "string",
  "Devices": [
    {
      "Id": "string",
      "ThingArn": "string",
      "CertificateArn": "string",
      "SyncShadow": "boolean"
    }
  ],
  "AmznClientToken": "string"
}

Responses

200 (CreateDeviceDefinitionVersionResponse)

VersionInformation (p. 521)

{
  "Arn": "string",
  "Id": "string",
  "Version": "string",
  "Version": "string",
  "CreationTimestamp": "string"
}

VersionInformation

Information about a version.

type: object

Arn

The ARN of the version.

type: string

Id

The ID of the version.

type: string

Version

The unique ID of the version.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the version was created.

type: string

400

invalid request

GeneralError (p. 476)

{
  "Message": "string",
}
"ErrorDetails": [
  {
    "DetailedErrorCode": "string",
    "DetailedErrorMessage": "string"
  }
]

GeneralError

  General error information.

  type: object

  required: ["Message"]

Message

  A message containing information about the error.

  type: string

ErrorDetails

  A list of error details.

  type: array

  items: ErrorDetail (p. 468)

ErrorDetail

  Details about the error.

  type: object

DetailedErrorCode

  A detailed error code.

  type: string

DetailedErrorMessage

  A detailed error message.

  type: string

**GET**

GET  /greengrass/definition/devices/DeviceDefinitionId/versions

Operation ID: ListDeviceDefinitionVersions (p. 157)

Lists the versions of a device definition.

Produces: application/json

Path Parameters

DeviceDefinitionId (p. 448)

  The ID of the device definition.

  where used: path; required: true
type: string

**Query Parameters**

**NextToken (p. 451)**

The token for the next set of results, or `null` if there are no additional results.

where used: query; required: false
type: string

**MaxResults (p. 450)**

The maximum number of results to be returned per request.

where used: query; required: false
type: integer

**CLI**

```
aws greengrass list-device-definition-versions \
  --device-definition-id <value>
  [--next-token <value>]
  [--max-results <value>]
  [--cli-input-json <value>]
  [--generate-cli-skeleton]
```

cli-input-json format:

```
{
  "DeviceDefinitionId": "string",
  "NextToken": "string",
  "MaxResults": "integer"
}
```

**Responses**

**200 (ListDeviceDefinitionVersionsResponse)**

**ListVersionsResponse (p. 502)**

```
{
  "Versions": [
    {
      "Arn": "string",
      "Id": "string",
      "Version": "string",
      "CreationTimestamp": "string"
    }
  ],
  "NextToken": "string"
}
```

ListVersionsResponse

A list of versions.
type: object

Versions

Information about a version.

type: array

items: VersionInformation (p. 521)

VersionInformation

Information about a version.

type: object

Arn

The ARN of the version.

type: string

Id

The ID of the version.

type: string

Version

The unique ID of the version.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the version was created.

type: string

NextToken

The token for the next set of results, or null if there are no additional results.

type: string

400

invalid request

GeneralError (p. 476)

{
   "Message": "string",
   "ErrorDetails": [
      {
         "DetailedErrorCode": "string",
         "DetailedErrorMessage": "string"
      }
   ]
}
AWS Greengrass API Reference
/greengrass/definition/devices/DeviceDefinitionId/
versions/DeviceDefinitionVersionId

required: ["Message"]
Message
A message containing information about the error.
  type: string
ErrorDetails
A list of error details.
  type: array
  items: ErrorDetail (p. 468)
ErrorDetail
Details about the error.
  type: object
DetailedErrorCode
A detailed error code.
  type: string
DetailedErrorMessage
A detailed error message.
  type: string

/greengrass/definition/devices/DeviceDefinitionId/
versions/DeviceDefinitionVersionId

GET
GET /greengrass/definition/devices/DeviceDefinitionId/
versions/DeviceDefinitionVersionId

Operation ID: GetDeviceDefinitionVersion (p. 108)

Retrieves information about a device definition version.

Produces: application/json

Path Parameters

DeviceDefinitionId (p. 448)
The ID of the device definition.
  where used: path; required: true
  type: string
DeviceDefinitionVersionId (p. 448)
The ID of the device definition version.
  where used: path; required: true
type: string

CLI

```bash
aws greengrass get-device-definition-version \
   --device-definition-id <value> \
   --device-definition-version-id <value> \
   [--cli-input-json <value>] \
   [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
   "DeviceDefinitionId": "string",
   "DeviceDefinitionVersionId": "string"
}
```

Responses

**200 (GetDeviceDefinitionVersionResponse)**

GetDeviceDefinitionVersionResponse (p. 481)

```json
{
   "Arn": "string",
   "Id": "string",
   "Version": "string",
   "CreationTimestamp": "string",
   "Definition": {
      "Devices": [
         {
            "Id": "string",
            "ThingArn": "string",
            "CertificateArn": "string",
            "SyncShadow": true
         }
      ]
   }
}
```

GetDeviceDefinitionVersionResponse

- type: object
- Arn
  The ARN of the device definition version.
  - type: string
- Id
  The ID of the device definition version.
  - type: string
- Version
  The version of the device definition version.
  - type: string
CreationTimestamp
   The time, in milliseconds since the epoch, when the device definition version was created.
   type: string

Definition
   Information about a device definition version.
   type: object

Devices
   A list of devices in the definition version.
   type: array
   items: Device (p. 465)

Device
   Information about a device.
   type: object
   required: ["Id", "ThingArn", "CertificateArn"]

Id
   The ID of the device.
   type: string

ThingArn
   The thing ARN of the device.
   type: string

CertificateArn
   The ARN of the certificate associated with the device.
   type: string

SyncShadow
   If true, the device's local shadow will be automatically synced with the cloud.
   type: boolean

400
   invalid request

GeneralError (p. 476)

```json
{
   "Message": "string",
   "ErrorDetails": [
   {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
   }
   ]
}
```
GeneralError
General error information.
type: object
required: ["Message"]
Message
A message containing information about the error.
type: string
ErrorDetails
A list of error details.
type: array
items: ErrorDetail (p. 468)
ErrorDetail
Details about the error.
type: object
DetailedErrorCode
A detailed error code.
type: string
DetailedErrorMessage
A detailed error message.
type: string

/greengrass/definition/functions

GET
GET /greengrass/definition/functions
Operation ID: ListFunctionDefinitions (p. 164)
Retrieves a list of Lambda function definitions.
Produces: application/json
Query Parameters
MaxResults (p. 450)
The maximum number of results to be returned per request.
where used: query; required: false
type: integer
NextToken (p. 451)
The token for the next set of results, or null if there are no additional results.
CLI

```bash
aws greengrass list-function-definitions
  [--max-results <value>] 
  [--next-token <value>] 
  [--cli-input-json <value>] 
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "MaxResults": "integer",
  "NextToken": "string"
}
```

Responses

**200 (ListFunctionDefinitionsResponse)**

Success. The response contains the IDs of all the Greengrass Lambda function definitions in this account.

ListDefinitionsResponse (p. 498)

```json
{
  "Definitions": [ 
    {
      "Name": "string",
      "Id": "string",
      "Arn": "string",
      "LastUpdatedTimestamp": "string",
      "CreationTimestamp": "string",
      "LatestVersion": "string",
      "LatestVersionArn": "string"
    },
    "NextToken": "string"
  ]
}
```

ListDefinitionsResponse

A list of definitions.

Definitions

Information about a definition.

type: object

Definitions

Information about a definition.
POST

POST /greengrass/definition/functions

Operation ID: CreateFunctionDefinition (p. 25)

Creates a Lambda function definition which contains a list of Lambda functions and their configurations to be used in a group. You can create an initial version of the definition by providing a list of Lambda functions and their configurations now, or use CreateFunctionDefinitionVersion later.

Produces: application/json

Body Parameters

CreateFunctionDefinitionRequestBody (p. 434)

where used: body; required: true
{
  "Name": "string",
  "InitialVersion": {
    "Functions": [
      {
        "Id": "string",
        "FunctionArn": "string",
        "FunctionConfiguration": {
          "Pinned": true,
          "Executable": "string",
          "ExecArgs": "string",
          "MemorySize": 0,
          "Timeout": 0,
          "EncodingType": "binary|json",
          "Environment": {
            "Variables": {
              "additionalProperty0": "string",
              "additionalProperty1": "string",
              "additionalProperty2": "string"
            },
            "ResourceAccessPolicies": [
              {
                "ResourceId": "string",
                "Permission": "ro|rw"
              }
            ],
            "AccessSysfs": true
          }
        }
      }
    ]
  }
}

**Name**

The name of the function definition.

type: string

**InitialVersion**

Information about a function definition version.

type: object

**Functions**

A list of Lambda functions in this function definition version.

type: array

items: Function (p. 469)

**Functions**

Information about a Lambda function.

type: object

required: ["Id"]

**Id**

The ID of the Lambda function.

type: string
FunctionArn

The ARN of the Lambda function.

type: string

FunctionConfiguration

The configuration of the Lambda function.

type: object

Pinned

True if the function is pinned. Pinned means the function is long-lived and starts when the core starts.

type: boolean

Executable

The name of the function executable.

type: string

ExecArgs

The execution arguments.

type: string

MemorySize

The memory size, in KB, which the function requires.

type: integer

Timeout

The allowed function execution time, after which Lambda should terminate the function. This timeout still applies to pinned lambdas for each request.

type: integer

EncodingType

The expected encoding type of the input payload for the function. The default is json.

type: string

enum: ["binary", "json"]

Environment

The environment configuration of the function.

type: object

Variables

Environment variables for the Lambda function's configuration.

type: object

additionalProperties: "type"=>"string"

ResourceAccessPolicies

A list of the resources, with their permissions, to which the Lambda function will be granted access. A Lambda function can have at most 10 resources.
type: array
items: ResourceAccessPolicy (p. 511)

ResourceAccessPolicy

A policy used by the function to access a resource.

type: object

required: ["ResourceId"]

ResourceId

The ID of the resource. (This ID is assigned to the resource when you create the resource definition.)

type: string

Permission

The type of permission a function has to access a resource.

type: string

enum: ["ro", "rw"]

AccessSysfs

If true, the Lambda function is allowed to access the host's /sys folder. Use this when the Lambda function needs to read device information from /sys.

type: boolean

Header Parameters

X-Amzn-Client-Token (p. 455)

A client token used to correlate requests and responses.

where used: header; required: false

type: string

CLI

```bash
aws greengrass create-function-definition \
  [--name <value>] \
  [--initial-version <value>] \
  [--amzn-client-token <value>] \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "Name": "string",
  "InitialVersion": {
    "Functions": [
      {
        "Id": "string",
        "FunctionArn": "string",
```
"FunctionConfiguration": {
    "Pinned": "boolean",
    "Executable": "string",
    "ExecArgs": "string",
    "MemorySize": "integer",
    "Timeout": "integer",
    "EncodingType": "binary|json",
    "Environment": {
        "Variables": {
            "additionalProperty0": "string",
            "additionalProperty1": "string",
            "additionalProperty2": "string"
        },
        "ResourceAccessPolicies": [
            {
                "ResourceId": "string",
                "Permission": "ro|rw"
            }
        ],
        "AccessSysfs": "boolean"
    }
},
"AmznClientToken": "string"

Responses

200 (CreateFunctionDefinitionResponse)

DefinitionInformation (p. 463)

{ "Name": "string",
  "Id": "string",
  "Arn": "string",
  "LastUpdatedTimestamp": "string",
  "CreationTimestamp": "string",
  "LatestVersion": "string",
  "LatestVersionArn": "string" }

DefinitionInformation

Information about a definition.

type: object

Name

The name of the definition.

type: string

Id

The ID of the definition.

type: string

Arn

The ARN of the definition.
type: string
LastUpdatedTimestamp
The time, in milliseconds since the epoch, when the definition was last updated.

type: string
CreationTimestamp
The time, in milliseconds since the epoch, when the definition was created.

type: string
LatestVersion
The latest version of the definition.

type: string
LatestVersionArn
The ARN of the latest version of the definition.

type: string
400
invalid request
GeneralError (p. 476)

{  
  "Message": "string",
  "ErrorDetails": [
    
    "DetailedErrorCode": "string",
    "DetailedErrorMessage": "string"
  ]
}

GeneralError
General error information.

type: object
required: ["Message"]
Message
A message containing information about the error.

type: string
ErrorDetails
A list of error details.

type: array
items: ErrorDetail (p. 468)

ErrorDetail
Details about the error.
AWS Greengrass API Reference
/greengrass/definition/functions/FunctionDefinitionId

```javascript
type: object
DetailedErrorCode
  A detailed error code.
  type: string
DetailedErrorMessage
  A detailed error message.
  type: string
```

/greengrass/definition/functions/FunctionDefinitionId

GET

GET /greengrass/definition/functions/FunctionDefinitionId

Operation ID: GetFunctionDefinition (p. 111)

Retrieves information about a Lambda function definition, including its creation time and latest version.

Produces: application/json

Path Parameters

FunctionDefinitionId (p. 449)

The ID of the Lambda function definition.

where used: path; required: true

type: string

CLI

```bash
aws greengrass get-function-definition \\
  --function-definition-id <value> \\
  [--cli-input-json <value>] \\
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "FunctionDefinitionId": "string"
}
```

Responses

200 (GetFunctionDefinitionResponse)

  success

    DefinitionInformation (p. 463)
DefinitionInformation

Information about a definition.

type: object

Name

The name of the definition.

type: string

Id

The ID of the definition.

type: string

Arn

The ARN of the definition.

type: string

LastUpdatedTimestamp

The time, in milliseconds since the epoch, when the definition was last updated.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the definition was created.

type: string

LatestVersion

The latest version of the definition.

type: string

LatestVersionArn

The ARN of the latest version of the definition.

type: string

400

invalid request

GeneralError (p. 476)
{  
  "DetailedErrorCode": "string",
  "DetailedErrorMessage": "string"
}
]

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

DELETE

DELETE /greengrass/definition/functions/FunctionDefinitionId

Operation ID: DeleteFunctionDefinition (p. 79)

Deletes a Lambda function definition.

Produces: application/json

Path Parameters

FunctionDefinitionId (p. 449)

The ID of the Lambda function definition.

where used: path; required: true
type: string

**CLI**

```bash
aws greengrass delete-function-definition \
   --function-definition-id <value> \
   [--cli-input-json <value>] \
   [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
   "FunctionDefinitionId": "string"
}
```

**Responses**

**200**

success

*Empty (p. 467)*

```json
{}
```

*Empty Schema*

- type: object

**400**

invalid request

*GeneralError (p. 476)*

```json
{
   "Message": "string",
   "ErrorDetails": [
      {
         "DetailedErrorCode": "string",
         "DetailedErrorMessage": "string"
      }
   ]
}
```

*GeneralError*

General error information.

- type: object

  required: ["Message"]

*Message*

A message containing information about the error.
**PUT**

PUT /greengrass/definition/functions/FunctionDefinitionId

Operation ID: UpdateFunctionDefinition (p. 199)

Updates a Lambda function definition.

Produces: application/json

**Body Parameters**

UpdateDefinitionRequestBody (p. 454)

where used: body; required: true

```json
{
  "Name": "string"
}
```

**Path Parameters**

FunctionDefinitionId (p. 449)

The ID of the Lambda function definition.
where used: path; required: true

type: string

CLI

```bash
aws greengrass update-function-definition \
  --function-definition-id <value> \
  [--name <value>]  \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "FunctionDefinitionId": "string",
  "Name": "string"
}
```

Responses

200

success

Empty (p. 467)

```json
{}
```

Empty Schema

Empty
type: object

400

invalid request

GeneralError (p. 476)

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

GeneralError

General error information.
type: object

required: ["Message"]
Message
A message containing information about the error.
type: string

ErrorDetails
A list of error details.
type: array
items: ErrorDetail (p. 468)

ErrorDetail
Details about the error.
type: object

DetailedErrorCode
A detailed error code.
type: string

DetailedErrorMessage
A detailed error message.
type: string

/greengrass/definition/functions/
FunctionDefinitionId/versions

POST
POST /greengrass/definition/functions/FunctionDefinitionId/versions

Operation ID: CreateFunctionDefinitionVersion (p. 30)

Creates a version of a Lambda function definition that has already been defined.

Produces: application/json

Body Parameters

CreateFunctionDefinitionVersionRequestBody (p. 435)

Information about the function definition version.

where used: body; required: true

{   "Functions": [   {     "Id": "string",     "FunctionArn": "string",     "FunctionConfiguration": {     "Pinned": true,     "Executable": "string",     "ExecArgs": "string",   }   }  ]
"MemorySize": 0,
"Timeout": 0,
"EncodingType": "binary|json",
"Environment": {
  "Variables": {
    "additionalProperty0": "string",
    "additionalProperty1": "string",
    "additionalProperty2": "string"
  },
  "ResourceAccessPolicies": [
    {
      "ResourceId": "string",
      "Permission": "ro|rw"
    }
  ],
  "AccessSysfs": true
}
}
]

FunctionDefinitionVersion

  Information about a function definition version.
  type: object

Functions

  A list of Lambda functions in this function definition version.
  type: array

  items: Function (p. 469)

  Information about a Lambda function.
  type: object
  required: ["Id"]

Id

  The ID of the Lambda function.
  type: string

FunctionArn

  The ARN of the Lambda function.
  type: string

FunctionConfiguration

  The configuration of the Lambda function.
  type: object

Pinned

  True if the function is pinned. Pinned means the function is long-lived and starts when the core starts.
  type: boolean
Executable
   The name of the function executable.
   type: string

ExecArgs
   The execution arguments.
   type: string

MemorySize
   The memory size, in KB, which the function requires.
   type: integer

Timeout
   The allowed function execution time, after which Lambda should terminate the function. This timeout still applies to pinned lambdas for each request.
   type: integer

EncodingType
   The expected encoding type of the input payload for the function. The default is json.
   type: string
   enum: ["binary", "json"]

Environment
   The environment configuration of the function.
   type: object

Variables
   Environment variables for the Lambda function's configuration.
   type: object
   additionalProperties: type=>string

ResourceAccessPolicies
   A list of the resources, with their permissions, to which the Lambda function will be granted access. A Lambda function can have at most 10 resources.
   type: array
   items: ResourceAccessPolicy (p. 511)

ResourceAccessPolicy
   A policy used by the function to access a resource.
   type: object
   required: ["ResourceId"]

ResourceId
   The ID of the resource. (This ID is assigned to the resource when you create the resource definition.)
type: string
Permission

The type of permission a function has to access a resource.

type: string
enum: ["ro", "rw"]

AccessSysfs

If true, the Lambda function is allowed to access the host's /sys folder. Use this when the Lambda function needs to read device information from /sys.

type: boolean

Header Parameters

X-Amzn-Client-Token (p. 455)

A client token used to correlate requests and responses.

where used: header; required: false

type: string

Path Parameters

FunctionDefinitionId (p. 449)

The ID of the Lambda function definition.

where used: path; required: true

type: string

CLI

```
aws greengrass create-function-definition-version \
    --function-definition-id <value> \
    [--functions <value>] \
    [--amzn-client-token <value>] \
    [--cli-input-json <value>] \
    [--generate-cli-skeleton]
```

cli-input-json format:

```
{
    "FunctionDefinitionId": "string",
    "Functions": [
        {
            "Id": "string",
            "FunctionArn": "string",
            "FunctionConfiguration": {
                "Pinned": "boolean",
                "Executable": "string",
                "ExecArgs": "string",
                "MemorySize": "integer",
            }
        }
    ]
}```
"Timeout": "integer",
"EncodingType": "binary|json",
"Environment": {
  "Variables": {
    "additionalProperty0": "string",
    "additionalProperty1": "string",
    "additionalProperty2": "string"
  },
  "ResourceAccessPolicies": [
    {
      "ResourceId": "string",
      "Permission": "ro|rw"
    }
  ],
  "AccessSysfs": "boolean"
},
"AmznClientToken": "string"

Responses

200 (CreateFunctionDefinitionVersionResponse)

VersionInformation (p. 521)

{
  "Arn": "string",
  "Id": "string",
  "Version": "string",
  "CreationTimestamp": "string"
}

VersionInformation

Information about a version.

type: object

Arn

The ARN of the version.

type: string

Id

The ID of the version.

type: string

Version

The unique ID of the version.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the version was created.

type: string
400

invalid request

GeneralError (p. 476)

```json
{
    "Message": "string",
    "ErrorDetails": [
        {
            "DetailedErrorCode": "string",
            "DetailedErrorMessage": "string"
        }
    ]
}
```

GeneralError

General error information.

- **type**: object
- **required**: ["Message"]

Message

A message containing information about the error.

- **type**: string

ErrorDetails

A list of error details.

- **type**: array
  - **items**: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

- **type**: object

DetailedErrorCode

A detailed error code.

- **type**: string

DetailedErrorMessage

A detailed error message.

- **type**: string

**GET**

GET /greengrass/definition/functions/FunctionDefinitionId/versions

Operation ID: ListFunctionDefinitionVersions (p. 162)

Lists the versions of a Lambda function definition.
Produce: application/json

Path Parameters

FunctionDefinitionId (p. 449)

The ID of the Lambda function definition.

where used: path; required: true

type: string

Query Parameters

NextToken (p. 451)

The token for the next set of results, or null if there are no additional results.

where used: query; required: false

type: string

MaxResults (p. 450)

The maximum number of results to be returned per request.

where used: query; required: false

type: integer

CLI

```
aws greengrass list-function-definition-versions \
  --function-definition-id <value> \
  [--next-token <value>] \
  [--max-results <value>] \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```
{
  "FunctionDefinitionId": "string",
  "NextToken": "string",
  "MaxResults": "integer"
}
```

Responses

200 (ListFunctionDefinitionVersionsResponse)

ListVersionsResponse (p. 502)

```
{
  "Versions": [
    {
      "Arn": "string",
      "Id": "string",
    }...
```

ListVersionsResponse

A list of versions.

type: object

Versions

Information about a version.

type: array

items: VersionInformation (p. 521)

VersionInformation

Information about a version.

type: object

Arn

The ARN of the version.

type: string

Id

The ID of the version.

type: string

Version

The unique ID of the version.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the version was created.

type: string

NextToken

The token for the next set of results, or null if there are no additional results.

type: string

400

invalid request

GeneralError (p. 476)

{
    "Message": "string",
    "ErrorDetails": [

GET

GET /greengrass/definition/functions/FunctionDefinitionId/versions/FunctionDefinitionVersionId

Operation ID: GetFunctionDefinitionVersion (p. 113)

Retrieves information about a Lambda function definition version, including which Lambda functions are included in the version and their configurations.
Path Parameters

FunctionDefinitionId (p. 449)

The ID of the Lambda function definition.

where used: path; required: true

type: string

FunctionDefinitionVersionId (p. 449)

The ID of the function definition version.

where used: path; required: true

type: string

CLI

```
aws greengrass get-function-definition-version \
  --function-definition-id <value> \
  --function-definition-version-id <value> \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "FunctionDefinitionId": "string",
  "FunctionDefinitionVersionId": "string"
}
```

Responses

200

success

GetFunctionDefinitionVersionResponse (p. 482)

```json
{
  "Arn": "string",
  "Id": "string",
  "Version": "string",
  "CreationTimestamp": "string",
  "Definition": {
    "Functions": [
      {
        "Id": "string",
        "FunctionArn": "string",
        "FunctionConfiguration": {
          "Pinned": true,
          "Executable": "string",
          "ExecArgs": "string",
          "MemorySize": 0,
          "Timeout": 0,
          "EncodingType": "binary|json",
```
"Environment": {
  "Variables": {
    "additionalProperty0": "string",
    "additionalProperty1": "string",
    "additionalProperty2": "string"
  },
  "ResourceAccessPolicies": [
    {
      "ResourceId": "string",
      "Permission": "ro|rw"
    }
  ],
  "AccessSysfs": true
}
}
]
}
]
}

GetFunctionDefinitionVersionResponse

Information about a function definition version.

type: object

Arn

The ARN of the function definition version.

type: string

Id

The ID of the function definition version.

type: string

Version

The version of the function definition version.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the function definition version was created.

type: string

Definition

Information about a function definition version.

type: object

Functions

A list of Lambda functions in this function definition version.

type: array

items: Function (p. 469)

Information about a Lambda function.

type: object
required: ["Id"]

**Id**

The ID of the Lambda function.

type: string

**FunctionArn**

The ARN of the Lambda function.

type: string

**FunctionConfiguration**

The configuration of the Lambda function.

type: object

**Pinned**

True if the function is pinned. Pinned means the function is long-lived and starts when the core starts.

type: boolean

**Executable**

The name of the function executable.

type: string

**ExecArgs**

The execution arguments.

type: string

**MemorySize**

The memory size, in KB, which the function requires.

type: integer

**Timeout**

The allowed function execution time, after which Lambda should terminate the function. This timeout still applies to pinned lambdas for each request.

type: integer

**EncodingType**

The expected encoding type of the input payload for the function. The default is json.

type: string

enum: ["binary", "json"]

**Environment**

The environment configuration of the function.

type: object

**Variables**

Environment variables for the Lambda function's configuration.
type: object

additionalProperties: "type"=>"string"

ResourceAccessPolicies

A list of the resources, with their permissions, to which the Lambda function will be granted access. A Lambda function can have at most 10 resources.

- type: array

  items: ResourceAccessPolicy (p. 511)

ResourceAccessPolicy

A policy used by the function to access a resource.

- type: object

  required: ["ResourceId"]

ResourceId

The ID of the resource. (This ID is assigned to the resource when you create the resource definition.)

- type: string

Permission

The type of permission a function has to access a resource.

- type: string

  enum: ["ro", "rw"]

AccessSysfs

If true, the Lambda function is allowed to access the host's /sys folder. Use this when the Lambda function needs to read device information from /sys.

- type: boolean

400

invalid request

GeneralError (p. 476)

{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}

GeneralError

General error information.

- type: object
required: ["Message"]
Message
A message containing information about the error.
type: string
ErrorDetails
A list of error details.
type: array
items: ErrorDetail (p. 468)
ErrorDetail
Details about the error.
type: object
DetailedErrorCode
A detailed error code.
type: string
DetailedErrorMessage
A detailed error message.
type: string

/greengrass/definition/loggers

GET
GET /greengrass/definition/loggers
Operation ID: ListLoggerDefinitions (p. 177)
Retrieves a list of logger definitions.
Produces: application/json

Query Parameters

MaxResults (p. 450)
The maximum number of results to be returned per request.
where used: query; required: false
type: integer

NextToken (p. 451)
The token for the next set of results, or null if there are no additional results.
where used: query; required: false
type: string
CLI

aws greengrass list-logger-definitions \
  [--max-results <value>] \
  [--next-token <value>] \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]

cli-input-json format:

{
  "MaxResults": "integer",
  "NextToken": "string"
}

Responses

200 (ListLoggerDefinitionsResponse)

ListDefinitionsResponse (p. 498)

{
  "Definitions": [
    {
      "Name": "string",
      "Id": "string",
      "Arn": "string",
      "LastUpdatedTimestamp": "string",
      "CreationTimestamp": "string",
      "LatestVersion": "string",
      "LatestVersionArn": "string"
    }
  ],
  "NextToken": "string"
}

ListDefinitionsResponse

A list of definitions.

Definitions

Information about a definition.

type: array

items: DefinitionInformation (p. 463)

DefinitionInformation

Information about a definition.

Name

The name of the definition.

type: string
Id
The ID of the definition.
type: string

Arn
The ARN of the definition.
type: string

LastUpdatedTimestamp
The time, in milliseconds since the epoch, when the definition was last updated.
type: string

CreationTimestamp
The time, in milliseconds since the epoch, when the definition was created.
type: string

LatestVersion
The latest version of the definition.
type: string

LatestVersionArn
The ARN of the latest version of the definition.
type: string

NextToken
The token for the next set of results, or null if there are no additional results.
type: string

**POST**

POST /greengrass/definition/loggers

Operation ID: CreateLoggerDefinition (p. 45)

Creates a logger definition. You may provide the initial version of the logger definition now or use CreateLoggerDefinitionVersion at a later time.

Produces: application/json

Body Parameters

CreateLoggerDefinitionRequestBody (p. 439)

where used: body; required: true

```json
{
    "Name": "string",
    "InitialVersion": {
        "Loggers": [
            {}
        ]
    }
}
```
"Id": "string",
"Type": "FileSystem|AWSCloudWatch",
"Component": "GreengrassSystem|Lambda",
"Level": "DEBUG|INFO|WARN|ERROR|FATAL",
"Space": 0
}
]
}
}

Name
The name of the logger definition.

type: string

InitialVersion
Information about a logger definition version.

type: object

Loggers
A list of loggers.

type: array

items: Logger (p. 505)

Logger
Information about a logger

type: object

required: ["Id", "Type", "Component", "Level"]

Id
The id of the logger.

type: string

Type

type: string

enum: ["FileSystem", "AWSCloudWatch"]

Component

type: string

enum: ["GreengrassSystem", "Lambda"]

Level

type: string

enum: ["DEBUG", "INFO", "WARN", "ERROR", "FATAL"]

Space
The amount of file space, in KB, to use if the local file system is used for logging purposes.

type: integer
Header Parameters

X-Amzn-Client-Token (p. 455)

A client token used to correlate requests and responses.

where used: header; required: false
type: string

CLI

```
aws greengrass create-logger-definition \
    [--name <value>] \ 
    [--initial-version <value>] \ 
    [--amzn-client-token <value>] \ 
    [--cli-input-json <value>] \ 
    [--generate-cli-skeleton]
```

cli-input-json format:

```
{
    "Name": "string",
    "InitialVersion": {
        "Loggers": [
            {
                "Id": "string",
                "Type": "FileSystem|AWSCloudWatch",
                "Component": "GreengrassSystem|Lambda",
                "Level": "DEBUG|INFO|WARN|ERROR|FATAL",
                "Space": "integer"
            }
        ],
    },
    "AmznClientToken": "string"
}
```

Responses

200 (CreateLoggerDefinitionResponse)

```
DefinitionInformation (p. 463)
```

```
{
    "Name": "string",
    "Id": "string",
    "Arn": "string",
    "LastUpdatedTimestamp": "string",
    "CreationTimestamp": "string",
    "LatestVersion": "string",
    "LatestVersionArn": "string"
}
```

DefinitionInformation

Information about a definition.
type: object
Name
The name of the definition.
type: string
Id
The ID of the definition.
type: string
Arn
The ARN of the definition.
type: string
LastUpdatedTimestamp
The time, in milliseconds since the epoch, when the definition was last updated.
type: string
CreationTimestamp
The time, in milliseconds since the epoch, when the definition was created.
type: string
LatestVersion
The latest version of the definition.
type: string
LatestVersionArn
The ARN of the latest version of the definition.
type: string

400
invalid request

**GeneralError (p. 476)**

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

**GeneralError**
General error information.
type: object
required: ["Message"]

**Message**
A message containing information about the error.
type: string
ErrorDetails
A list of error details.
type: array
items: ErrorDetail (p. 468)
ErrorDetail
Details about the error.
type: object
DetailedErrorCode
A detailed error code.
type: string
DetailedErrorMessage
A detailed error message.
type: string

/greengrass/definition/loggers/LoggerDefinitionId
GET
GET /greengrass/definition/loggers/LoggerDefinitionId
Operation ID: GetLoggerDefinition (p. 129)
Retrieves information about a logger definition.
Produces: application/json
Path Parameters
LoggerDefinitionId (p. 450)
The ID of the logger definition.
where used: path; required: true
type: string

CLI

aws greengrass get-logger-definition \
--logger-definition-id <value> \
[--cli-input-json <value>] \
[--generate-cli-skeleton]

cli-input-json format:

{

"LoggerDefinitionId": "string"
}

Responses

200 (GetLoggerDefinitionResponse)

DefinitionInformation (p. 463)

{
   "Name": "string",
   "Id": "string",
   "Arn": "string",
   "LastUpdatedTimestamp": "string",
   "CreationTimestamp": "string",
   "LatestVersion": "string",
   "LatestVersionArn": "string"
}

DefinitionInformation

Information about a definition.

type: object

Name

The name of the definition.

type: string

Id

The ID of the definition.

type: string

Arn

The ARN of the definition.

type: string

LastUpdatedTimestamp

The time, in milliseconds since the epoch, when the definition was last updated.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the definition was created.

type: string

LatestVersion

The latest version of the definition.

type: string

LatestVersionArn

The ARN of the latest version of the definition.

type: string
400
invalid request

GeneralError (p. 476)

{  
"Message": "string",
"ErrorDetails": [
  
  {  
    "DetailedErrorCode": "string",
    "DetailedErrorMessage": "string"
  }
]
}

GeneralError
General error information.
type: object
required: ["Message"]

Message
A message containing information about the error.
type: string

ErrorDetails
A list of error details.
type: array
items: ErrorDetail (p. 468)

ErrorDetail
Details about the error.
type: object

DetailedErrorCode
A detailed error code.
type: string

DetailedErrorMessage
A detailed error message.
type: string

DELETE

DELETE /greengrass/definition/loggers/LoggerDefinitionId

Operation ID: DeleteLoggerDefinition (p. 82)

Deletes a logger definition.

Produces: application/json
Path Parameters

**LoggerDefinitionId** (p. 450)

The ID of the logger definition.

where used: path; required: true
type: string

CLI

```bash
aws greengrass delete-logger-definition \
  --logger-definition-id <value> \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "LoggerDefinitionId": "string"
}
```

Responses

**200**
success

**Empty (p. 467)**

```json
{
}
```

Empty Schema

Empty
type: object

**400**
invalid request

**GeneralError (p. 476)**

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

GeneralError

General error information.
type: object
required: ["Message"]
Message
A message containing information about the error.
type: string
ErrorDetails
A list of error details.
type: array
items: ErrorDetail (p. 468)
ErrorDetail
Details about the error.
type: object
DetailedErrorCode
A detailed error code.
type: string
DetailedErrorMessage
A detailed error message.
type: string

PUT

PUT /greengrass/definition/loggers/LoggerDefinitionId

Operation ID: UpdateLoggerDefinition (p. 206)

Updates a logger definition.

Produces: application/json

Body Parameters

UpdateDefinitionRequestBody (p. 454)

where used: body; required: true

{"Name": "string"}

Name
The name of the definition.
required: true
type: string
Path Parameters

LoggerDefinitionId (p. 450)

The ID of the logger definition.
where used: path; required: true
type: string

CLI

```
aws greengrass update-logger-definition \
  --logger-definition-id <value> \
  [--name <value>] \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```
{
  "LoggerDefinitionId": "string",
  "Name": "string"
}
```

Responses

200

success

Empty (p. 467)

```
{
}
```

Empty Schema

Empty
type: object

400

invalid request

GeneralError (p. 476)

```
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```
GeneralError
  General error information.
  type: object
  required: ["Message"]
Message
  A message containing information about the error.
  type: string
ErrorDetails
  A list of error details.
  type: array
  items: ErrorDetail (p. 468)
ErrorDetail
  Details about the error.
  type: object
DetailedErrorCode
  A detailed error code.
  type: string
DetailedErrorMessage
  A detailed error message.
  type: string

/greengrass/definition/loggers/LoggerDefinitionId/versions

POST

POST /greengrass/definition/loggers/LoggerDefinitionId/versions

Operation ID: CreateLoggerDefinitionVersion (p. 49)

Creates a version of a logger definition that has already been defined.

Produces: application/json

Body Parameters

CreateLoggerDefinitionVersionRequestBody (p. 440)

where used: body; required: true

```json
{
  "Loggers": [
    
  ]
```

295
LoggerDefinitionVersion

Information about a logger definition version.

type: object

Loggers

A list of loggers.

type: array

items: Logger (p. 505)

Logger

Information about a logger

type: object

required: ["Id", "Type", "Component", "Level"]

Id

The id of the logger.

type: string

Type

type: string

enum: ["FileSystem", "AWSCloudWatch"]

Component

type: string

enum: ["GreengrassSystem", "Lambda"]

Level

type: string

enum: ["DEBUG", "INFO", "WARN", "ERROR", "FATAL"]

Space

The amount of file space, in KB, to use if the local file system is used for logging purposes.

type: integer

Header Parameters

X-Amzn-Client-Token (p. 455)

A client token used to correlate requests and responses.
Path Parameters

**LoggerDefinitionId (p. 450)**

The ID of the logger definition.

where used: path; required: true

type: string

### CLI

```bash
aws greengrass create-logger-definition-version
   --logger-definition-id <value> \
   [--loggers <value>] \
   [--amzn-client-token <value>] \
   [--cli-input-json <value>] \
   [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
    "LoggerDefinitionId": "string",
    "Loggers": [
        {
            "Id": "string",
            "Type": "FileSystem|AWSCloudWatch",
            "Component": "GreengrassSystem|Lambda",
            "Level": "DEBUG|INFO|WARN|ERROR|FATAL",
            "Space": "integer"
        }
    ],
    "AmznClientToken": "string"
}
```

### Responses

**200** (*CreateLoggerDefinitionVersionResponse*)

```
VersionInformation (p. 521)
```

```json
{
    "Arn": "string",
    "Id": "string",
    "Version": "string",
    "CreationTimestamp": "string"
}
```

VersionInformation

Information about a version.

type: object
Arn
The ARN of the version.
type: string
Id
The ID of the version.
type: string
Version
The unique ID of the version.
type: string
CreationTimestamp
The time, in milliseconds since the epoch, when the version was created.
type: string

400
invalid request

GeneralError (p. 476)

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

GeneralError
General error information.
type: object
required: ["Message"]
Message
A message containing information about the error.
type: string
ErrorDetails
A list of error details.
type: array
items: ErrorDetail (p. 468)

ErrorDetail
Details about the error.
type: object
DetailedErrorCode
A detailed error code.
type: string

DetailedErrorMessage
A detailed error message.
type: string

GET
GET /greengrass/definition/loggers/LoggerDefinitionId/versions

Operation ID: ListLoggerDefinitionVersions (p. 174)
Lists the versions of a logger definition.
Produces: application/json

Path Parameters
LoggerDefinitionId (p. 450)
The ID of the logger definition.
where used: path; required: true
type: string

Query Parameters
NextToken (p. 451)
The token for the next set of results, or null if there are no additional results.
where used: query; required: false
type: string

MaxResults (p. 450)
The maximum number of results to be returned per request.
where used: query; required: false
type: integer

CLI
aws greengrass list-logger-definition-versions
   --logger-definition-id <value> \n   [--next-token <value>] \n   [--max-results <value>] \n   [--cli-input-json <value>] \n   [--generate-cli-skeleton]
cli-input-json format:

```json
{
    "LoggerDefinitionId": "string",
    "NextToken": "string",
    "MaxResults": "integer"
}
```

Responses

**200** (ListLoggerDefinitionVersionsResponse)

*ListVersionsResponse (p. 502)*

```json
{
    "Versions": [
    {
        "Arn": "string",
        "Id": "string",
        "Version": "string",
        "CreationTimestamp": "string"
    }
    ],
    "NextToken": "string"
}
```

**ListVersionsResponse**

A list of versions.

*type: object*

**Versions**

Information about a version.

*type: array*

*items: VersionInformation (p. 521)*

**VersionInformation**

Information about a version.

*type: object*

**Arn**

The ARN of the version.

*type: string*

**Id**

The ID of the version.

*type: string*

**Version**

The unique ID of the version.

*type: string*
CreationTimestamp

The time, in milliseconds since the epoch, when the version was created.

type: string

NextToken

The token for the next set of results, or null if there are no additional results.

400

invalid request

GeneralError (p. 476)

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string
/greengrass/definition/loggers/LoggerDefinitionId/versions/LoggerDefinitionVersionId

GET

GET /greengrass/definition/loggers/LoggerDefinitionId/versions/LoggerDefinitionVersionId

Operation ID: GetLoggerDefinitionVersion (p. 132)

Retrieves information about a logger definition version.

Produces: application/json

Path Parameters

**LoggerDefinitionVersionId** (p. 450)

The ID of the logger definition version.

where used: path; required: true

type: string

**LoggerDefinitionId** (p. 450)

The ID of the logger definition.

where used: path; required: true

type: string

CLI

```
aws greengrass get-logger-definition-version \
   --logger-definition-version-id <value> \
   --logger-definition-id <value> \
   [--cli-input-json <value>] \
   [--generate-cli-skeleton]
```

cli-input-json format:

```
{
   "LoggerDefinitionVersionId": "string",
   "LoggerDefinitionId": "string"
}
```

Responses

**200** (GetLoggerDefinitionVersionResponse)

success

GetLoggerDefinitionVersionResponse (p. 487)

```
GetLoggerDefinitionVersionResponse

Information about a logger definition version.

type: object

Arn

The ARN of the logger definition version.

type: string

Id

The ID of the logger definition version.

type: string

Version

The version of the logger definition version.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the logger definition version was created.

type: string

Definition

Information about a logger definition version.

type: object

Loggers

A list of loggers.

type: array

items: Logger (p. 505)

Logger

Information about a logger

type: object

required: ["Id", "Type", "Component", "Level"]
Id
   The id of the logger.
   type: string
Type
   type: string
   enum: ["FileSystem", "AWSCloudWatch"]
Component
   type: string
   enum: ["GreengrassSystem", "Lambda"]
Level
   type: string
   enum: ["DEBUG", "INFO", "WARN", "ERROR", "FATAL"]
Space
   The amount of file space, in KB, to use if the local file system is used for logging purposes.
   type: integer

400
invalid request

GeneralError (p. 476)

{
   "Message": "string",
   "ErrorDetails": [
   {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
   }
   ]
}

GeneralError
   General error information.
   type: object
   required: ["Message"]
Message
   A message containing information about the error.
   type: string
ErrorDetails
   A list of error details.
   type: array
   items: ErrorDetail (p. 468)
ErrorDetail
   Details about the error.
   type: object
DetailedErrorCode
   A detailed error code.
   type: string
DetailedErrorMessage
   A detailed error message.
   type: string

/greengrass/definition/resources

GET
GET /greengrass/definition/resources
Operation ID: ListResourceDefinitions (p. 182)
Retrieves a list of resource definitions.
Produces: application/json

Query Parameters
MaxResults (p. 450)
   The maximum number of results to be returned per request.
   where used: query; required: false
   type: integer
NextToken (p. 451)
   The token for the next set of results, or null if there are no additional results.
   where used: query; required: false
   type: string

CLI

aws greengrass list-resource-definitions
   [--max-results <value>] \
   [--next-token <value>] \
   [--cli-input-json <value>] \
   [--generate-cli-skeleton]

cli-input-json format:

{

"MaxResults": "integer",
"NextToken": "string"
}

Responses

**200 (ListResourceDefinitionsResponse)**

The IDs of all the Greengrass resource definitions in this account.

*ListDefinitionsResponse (p. 498)*

```json
{
  "Definitions": [
    {
      "Name": "string",
      "Id": "string",
      "Arn": "string",
      "LastUpdatedTimestamp": "string",
      "CreationTimestamp": "string",
      "LatestVersion": "string",
      "LatestVersionArn": "string"
    }
  ],
  "NextToken": "string"
}
```

*ListDefinitionsResponse*

A list of definitions.

- **type**: object
- **Definitions**
  - **Information about a definition.**
  - **type**: array
  - **items**: *DefinitionInformation (p. 463)*
- **DefinitionInformation**
  - **Information about a definition.**
  - **type**: object
  - **Name**
    - The name of the definition.
    - **type**: string
  - **Id**
    - The ID of the definition.
    - **type**: string
  - **Arn**
    - The ARN of the definition.
    - **type**: string
LastUpdatedTimestamp

The time, in milliseconds since the epoch, when the definition was last updated.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the definition was created.

 type: string

LatestVersion

The latest version of the definition.

 type: string

LatestVersionArn

The ARN of the latest version of the definition.

 type: string

NextToken

The token for the next set of results, or null if there are no additional results.

 type: string

**POST**

**POST /greengrass/definition/resources**

**Operation ID:** CreateResourceDefinition (p. 52)

Creates a resource definition which contains a list of resources to be used in a group. You can create an initial version of the definition by providing a list of resources now, or use CreateResourceDefinitionVersion later.

**Produces:** application/json

**Body Parameters**

**CreateResourceDefinitionRequestBody** (p. 441)

where used: body; required: true

```json
{
    "Name": "string",
    "InitialVersion": {
        "Resources": [
            {
                "Id": "string",
                "Name": "string",
                "ResourceDataContainer": {
                    "**NOTE:** choose one of the following --**",
                    "LocalDeviceResourceData": {
                        "SourcePath": "string",
                        "GroupOwnerSetting": {
                            "AutoAddGroupOwner": true,
                            "GroupOwner": "string"
                        }
                    }
                }
            }
        ]
    }
}
```
Name

The name of the resource definition.

type: string

InitialVersion

Information about a resource definition version.

type: object

Resources

A list of resources.

type: array

items: Resource (p. 508)

Resource

Information about a resource.


type: object

required: ["Id", "Name", "ResourceDataContainer"]

Id

The resource ID, used to refer to a resource in the Lambda function configuration. Max length is 128 characters with pattern [(a-zA-Z0-9:_-\+)]. This must be unique within a Greengrass group.

type: string

Name

The descriptive resource name, which is displayed on the AWS Greengrass console. Max length 128 characters with pattern [(a-zA-Z0-9:_-\+)]. This must be unique within a Greengrass group.

type: string
ResourceDataContainer

A container for resource data. The container takes only one of the following supported resource data types: LocalDeviceResourceData, LocalVolumeResourceData, SageMakerMachineLearningModelResourceData, S3MachineLearningModelResourceData.

type: object

NOTE: oneOf

LocalDeviceResourceData

Attributes that define a local device resource.

type: object

SourcePath

The local absolute path of the device resource. The source path for a device resource can refer only to a character device or block device under /dev.

type: string

GroupOwnerSetting

Group owner related settings for local resources.

type: object

AutoAddGroupOwner

If true, AWS Greengrass automatically adds the specified Linux OS group owner of the resource to the Lambda process privileges. Thus the Lambda process will have the file access permissions of the added Linux group.

type: boolean

GroupOwner

The name of the Linux OS group whose privileges will be added to the Lambda process. This field is optional.

type: string

LocalVolumeResourceData

Attributes that define a local volume resource.

type: object

SourcePath

The local absolute path of the volume resource on the host. The source path for a volume resource type cannot start with /proc or /sys.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.

type: string

GroupOwnerSetting

Group owner related settings for local resources.

type: object
AutoAddGroupOwner

If true, AWS Greengrass automatically adds the specified Linux OS group owner of the resource to the Lambda process privileges. Thus the Lambda process will have the file access permissions of the added Linux group.

type: boolean

GroupOwner

The name of the Linux OS group whose privileges will be added to the Lambda process. This field is optional.

type: string

SageMakerMachineLearningModelResourceData

Attributes that define an Amazon SageMaker machine learning resource.

type: object

SageMakerJobArn

The ARN of the Amazon SageMaker training job that represents the source model.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.

type: string

S3MachineLearningModelResourceData

Attributes that define an Amazon S3 machine learning resource.

type: object

S3Uri

The URI of the source model in an Amazon S3 bucket. The model package must be in tar.gz or .zip format.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.

type: string

**Header Parameters**

**X-Amzn-Client-Token (p. 455)**

A client token used to correlate requests and responses.

where used: header; required: false

type: string

**CLI**

```bash
aws greengrass create-resource-definition \
```
cli-input-json format:

```json
{
  "Name": "string",
  "InitialVersion": {
    "Resources": [
      {
        "Id": "string",
        "Name": "string",
        "ResourceDataContainer": {
          "**-- NOTE: choose one of the following --**",
          "LocalDeviceResourceData": {
            "SourcePath": "string",
            "GroupOwnerSetting": {
              "AutoAddGroupOwner": "boolean",
              "GroupOwner": "string"
            }
          },
          "LocalVolumeResourceData": {
            "SourcePath": "string",
            "DestinationPath": "string",
            "GroupOwnerSetting": {
              "AutoAddGroupOwner": "boolean",
              "GroupOwner": "string"
            }
          },
          "SageMakerMachineLearningModelResourceData": {
            "SageMakerJobArn": "string",
            "DestinationPath": "string"
          },
          "S3MachineLearningModelResourceData": {
            "S3Uri": "string",
            "DestinationPath": "string"
          }
        }
      }
    ],
    "AmznClientToken": "string"
  }
}
```

Responses

**200** (CreateResourceDefinitionResponse)

**DefinitionInformation (p. 463)**

```
{
  "Name": "string",
  "Id": "string",
  "Arn": "string",
  "LastUpdatedTimestamp": "string",
  "CreationTimestamp": "string",
  "LatestVersion": "string",
  "LatestVersionArn": "string"
}
```
DefinitionInformation
   Information about a definition.
   type: object
Name
   The name of the definition.
   type: string
Id
   The ID of the definition.
   type: string
Arn
   The ARN of the definition.
   type: string
LastUpdatedTimestamp
   The time, in milliseconds since the epoch, when the definition was last updated.
   type: string
CreationTimestamp
   The time, in milliseconds since the epoch, when the definition was created.
   type: string
LatestVersion
   The latest version of the definition.
   type: string
LatestVersionArn
   The ARN of the latest version of the definition.
   type: string

400
invalid request

GeneralError (p. 476)

```json
{
   "Message": "string",
   "ErrorDetails": [
      {
         "DetailedErrorCode": "string",
         "DetailedErrorMessage": "string"
      }
   ]
}
```

GeneralError
   General error information.
   type: object
/greengrass/definition/resources/ResourceDefinitionId

GET

GET /greengrass/definition/resources/ResourceDefinitionId

Operation ID: GetResourceDefinition (p. 135)

Retrieves information about a resource definition, including its creation time and latest version.

Produces: application/json

Path Parameters

ResourceDefinitionId (p. 452)

- The ID of the resource definition.
  - where used: path; required: true
  - type: string

CLI

```
aws greengrass get-resource-definition
```
CLI Input:

```
--resource-definition-id <value> \
[--cli-input-json <value>] \ 
[--generate-cli-skeleton]
```

**cli-input-json format:**

```
{
  "ResourceDefinitionId": "string"
}
```

**Responses**

**200** (GetResourceDefinitionResponse)

- success
  - DefinitionInformation (p. 463)

```
{
  "Name": "string",
  "Id": "string",
  "Arn": "string",
  "LastUpdatedTimestamp": "string",
  "CreationTimestamp": "string",
  "LatestVersion": "string",
  "LatestVersionArn": "string"
}
```

**DefinitionInformation**

Information about a definition.

- **Name**
  - type: string
  - The name of the definition.

- **Id**
  - type: string
  - The ID of the definition.

- **Arn**
  - type: string
  - The ARN of the definition.

- **LastUpdatedTimestamp**
  - type: string
  - The time, in milliseconds since the epoch, when the definition was last updated.

- **CreationTimestamp**
  - type: string
  - The time, in milliseconds since the epoch, when the definition was created.
LatestVersion

The latest version of the definition.

type: string

LatestVersionArn

The ARN of the latest version of the definition.

type: string

400

invalid request

GeneralError (p. 476)

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string
DELETE

DELETE /greengrass/definition/resources/ResourceDefinitionId

Operation ID: DeleteResourceDefinition (p. 84)

Deletes a resource definition.

Produces: application/json

Path Parameters

ResourceDefinitionId (p. 452)

The ID of the resource definition.

where used: path; required: true

type: string

CLI

```
aws greengrass delete-resource-definition \\n  --resource-definition-id <value> \\n  [--cli-input-json <value>] \\n  [--generate-cli-skeleton]
```

cli-input-json format:

```
{   "ResourceDefinitionId": "string"
}
```

Responses

200

success

Empty (p. 467)

```
{   }
```

Empty Schema

Empty
type: object

400

invalid request

GeneralError (p. 476)

```
{
```

"Message": "string",
"ErrorDetails": [
  {
    "DetailedErrorCode": "string",
    "DetailedErrorMessage": "string"
  }
]

**GeneralError**

General error information.

**type**: object

**required**: ["Message"]

**Message**

A message containing information about the error.

**type**: string

**ErrorDetails**

A list of error details.

**type**: array

**items**: ErrorDetail (p. 468)

**ErrorDetail**

Details about the error.

**type**: object

**DetailedErrorCode**

A detailed error code.

**type**: string

**DetailedErrorMessage**

A detailed error message.

**type**: string

**PUT**

PUT /greengrass/definition/resources/ResourceDefinitionId

**Operation ID**: UpdateResourceDefinition (p. 208)

Updates a resource definition.

**Produces**: application/json

**Body Parameters**

**UpdateDefinitionRequestBody** (p. 454)

where used: body; required: true
Name
The name of the definition.
required: true
type: string

Path Parameters

ResourceDefinitionId (p. 452)
The ID of the resource definition.
where used: path; required: true
type: string

CLI

```bash
aws greengrass update-resource-definition \
  --resource-definition-id <value> \
  [--name <value>] \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "ResourceDefinitionId": "string",
  "Name": "string"
}
```

Responses
200
success
Empty (p. 467)

```json
{
}
```

Empty Schema
Empty
type: object

400
invalid request
GeneralError (p. 476)

```json
{
   "Message": "string",
   "ErrorDetails": [
   {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
   }
   ]
}
```

GeneralError
General error information.
type: object
required: ["Message"]

Message
A message containing information about the error.
type: string

ErrorDetails
A list of error details.
type: array
items: ErrorDetail (p. 468)

ErrorDetail
Details about the error.
type: object

DetailedErrorCode
A detailed error code.
type: string

DetailedErrorMessage
A detailed error message.
type: string

/greengrass/definition/resources/
ResourceDefinitionId/versions

POST

POST /greengrass/definition/resources/ResourceDefinitionId/versions

Operation ID: CreateResourceDefinitionVersion (p. 58)

Creates a version of a resource definition that has already been defined.
Body Parameters

CreateResourceDefinitionVersionRequestBody (p. 442)

Information about the resource definition version.

where used: body; required: true

```
{
    "Resources": [
    {
        "Id": "string",
        "Name": "string",
        "ResourceDataContainer": {
            "--- NOTE:": "choose one of the following --*",
            "LocalDeviceResourceData": {
                "SourcePath": "string",
                "GroupOwnerSetting": {
                    "AutoAddGroupOwner": true,
                    "GroupOwner": "string"
                }
            },
            "LocalVolumeResourceData": {
                "SourcePath": "string",
                "DestinationPath": "string",
                "GroupOwnerSetting": {
                    "AutoAddGroupOwner": true,
                    "GroupOwner": "string"
                }
            },
            "SageMakerMachineLearningModelResourceData": {
                "SageMakerJobArn": "string",
                "DestinationPath": "string"
            },
            "S3MachineLearningModelResourceData": {
                "S3Uri": "string",
                "DestinationPath": "string"
            }
        }
    }
    ]
}
```

ResourceDefinitionVersion

Information about a resource definition version.

- **type**: object

Resources

A list of resources.

- **type**: array

  - **items**: Resource (p. 508)

Resource

Information about a resource.

- **type**: object
required: ["Id", "Name", "ResourceDataContainer"]

Id
The resource ID, used to refer to a resource in the Lambda function configuration. Max length is 128 characters with pattern [a-zA-Z0-9_:\-]+. This must be unique within a Greengrass group.
type: string

Name
The descriptive resource name, which is displayed on the AWS Greengrass console. Max length 128 characters with pattern [a-zA-Z0-9_:\-]+. This must be unique within a Greengrass group.
type: string

ResourceDataContainer
A container for resource data. The container takes only one of the following supported resource data types: LocalDeviceResourceData, LocalVolumeResourceData, SageMakerMachineLearningModelResourceData, S3MachineLearningModelResourceData.
type: object

NOTE: oneOf

LocalDeviceResourceData
Attributes that define a local device resource.
type: object

SourcePath
The local absolute path of the device resource. The source path for a device resource can refer only to a character device or block device under /dev.
type: string

GroupOwnerSetting
Group owner related settings for local resources.
type: object

AutoAddGroupOwner
If true, AWS Greengrass automatically adds the specified Linux OS group owner of the resource to the Lambda process privileges. Thus the Lambda process will have the file access permissions of the added Linux group.
type: boolean

GroupOwner
The name of the Linux OS group whose privileges will be added to the Lambda process. This field is optional.
type: string

LocalVolumeResourceData
Attributes that define a local volume resource.
type: object
SourcePath

The local absolute path of the volume resource on the host. The source path for a volume resource type cannot start with /proc or /sys.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.

type: string

GroupOwnerSetting

Group owner related settings for local resources.

type: object

AutoAddGroupOwner

If true, AWS Greengrass automatically adds the specified Linux OS group owner of the resource to the Lambda process privileges. Thus the Lambda process will have the file access permissions of the added Linux group.

type: boolean

GroupOwner

The name of the Linux OS group whose privileges will be added to the Lambda process. This field is optional.

type: string

SageMakerMachineLearningModelResourceData

Attributes that define an Amazon SageMaker machine learning resource.

type: object

SageMakerJobArn

The ARN of the Amazon SageMaker training job that represents the source model.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.

type: string

S3MachineLearningModelResourceData

Attributes that define an Amazon S3 machine learning resource.

type: object

S3Uri

The URI of the source model in an Amazon S3 bucket. The model package must be in tar.gz or .zip format.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.
type: string

Header Parameters

X-Amzn-Client-Token (p. 455)
A client token used to correlate requests and responses.
where used: header; required: false
type: string

Path Parameters

ResourceDefinitionId (p. 452)
The ID of the resource definition.
where used: path; required: true
type: string

CLI

aws greengrass create-resource-definition-version \
   --resource-definition-id <value> \
   [--resources <value>] \
   [--amzn-client-token <value>] \
   [--cli-input-json <value>] \
   [--generate-cli-skeleton]

cli-input-json format:

{
   "ResourceDefinitionId": "string",
   "Resources": [
      {
         "Id": "string",
         "Name": "string",
         "ResourceDataContainer": {
            "*-- NOTE:": "choose one of the following --*",
            "LocalDeviceResourceData": {
               "SourcePath": "string",
               "GroupOwnerSetting": {
                  "AutoAddGroupOwner": "boolean",
                  "GroupOwner": "string"
               }
            },
            "LocalVolumeResourceData": {
               "SourcePath": "string",
               "DestinationPath": "string",
               "GroupOwnerSetting": {
                  "AutoAddGroupOwner": "boolean",
                  "GroupOwner": "string"
               }
            },
            "SageMakerMachineLearningModelResourceData": {
               "SageMakerJobArn": "string",
               "DestinationPath": "string"
            }
         }
      }
   ]
}


```
},
  "S3MachineLearningModelResourceData": {
    "S3Uri": "string",
    "DestinationPath": "string"
  }
},
  "AmznClientToken": "string"
}

**Responses**

**200 (CreateResourceDefinitionVersionResponse)**

**VersionInformation (p. 521)**

```json
{
  "Arn": "string",
  "Id": "string",
  "Version": "string",
  "CreationTimestamp": "string"
}
```

**VersionInformation**

Information about a version.

- **Arn**
  - The ARN of the version.
  - type: string

- **Id**
  - The ID of the version.
  - type: string

- **Version**
  - The unique ID of the version.
  - type: string

- **CreationTimestamp**
  - The time, in milliseconds since the epoch, when the version was created.
  - type: string

**400**

invalid request

**GeneralError (p. 476)**

```json
{
  "Message": "string",
  "ErrorDetails": [
    
  ]
}
```
GET /greengrass/definition/resources/{ResourceDefinitionId}/versions

Operation ID: ListResourceDefinitionVersions (p. 179)

Lists the versions of a resource definition.

Produces: application/json

**Path Parameters**

**ResourceDefinitionId** (p. 452)

The ID of the resource definition.

where used: path; required: true
type: string

**Query Parameters**

**NextToken (p. 451)**

The token for the next set of results, or null if there are no additional results.

where used: query; required: false

type: string

**MaxResults (p. 450)**

The maximum number of results to be returned per request.

where used: query; required: false

type: integer

**CLI**

```bash
aws greengrass list-resource-definition-versions \
  --resource-definition-id <value> \
  [--next-token <value>] \
  [--max-results <value>] \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "ResourceDefinitionId": "string",
  "NextToken": "string",
  "MaxResults": "integer"
}
```

**Responses**

**200** (ListResourceDefinitionVersionsResponse)

**ListVersionsResponse (p. 502)**

```json
{
  "Versions": [
    {
      "Arn": "string",
      "Id": "string",
      "Version": "string",
      "CreationTimestamp": "string"
    }
  ],
  "NextToken": "string"
}
```

ListVersionsResponse

A list of versions.
type: object

Versions

Information about a version.

type: array

items: VersionInformation (p. 521)

VersionInformation

Information about a version.

type: object

Arn

The ARN of the version.

type: string

Id

The ID of the version.

type: string

Version

The unique ID of the version.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the version was created.

type: string

NextToken

The token for the next set of results, or null if there are no additional results.

type: string

400

invalid request

GeneralError (p. 476)

{
    "Message": "string",
    "ErrorDetails": [
        {
            "DetailedErrorCode": "string",
            "DetailedErrorMessage": "string"
        }
    ]
}

GeneralError

General error information.

type: object
required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

---

**/greengrass/definition/resources/ResourceDefinitionId/versions/ResourceDefinitionVersionId**

**GET**

GET /greengrass/definition/resources/ResourceDefinitionId/versions/ResourceDefinitionVersionId

**Operation ID:** GetResourceDefinitionVersion (p. 137)

Retrieves information about a resource definition version, including which resources are included in the version.

**Produces:** application/json

**Path Parameters**

**ResourceDefinitionId** (p. 452)

The ID of the resource definition.

where used: path; required: true

type: string
ResourceDefinitionVersionId (p. 452)

The ID of the resource definition version.

  where used: path; required: true

  type: string

CLI

```bash
aws greengrass get-resource-definition-version \
  --resource-definition-id <value> \
  --resource-definition-version-id <value> \
  [--cli-input-json <value>]  \
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "ResourceDefinitionId": "string",
  "ResourceDefinitionVersionId": "string"
}
```

Responses

200

success

GetResourceDefinitionVersionResponse (p. 489)

```json
{
  "Arn": "string",
  "Id": "string",
  "Version": "string",
  "CreationTimestamp": "string",
  "Definition": {
    "Resources": [
      {
        "Id": "string",
        "Name": "string",
        "ResourceDataContainer": {
          "-- NOTE:": "choose one of the following --*",
          "LocalDeviceResourceData": {
            "SourcePath": "string",
            "GroupOwnerSetting": {
              "AutoAddGroupOwner": true,
              "GroupOwner": "string"
            }
          },
          "LocalVolumeResourceData": {
            "SourcePath": "string",
            "DestinationPath": "string",
            "GroupOwnerSetting": {
              "AutoAddGroupOwner": true,
              "GroupOwner": "string"
            }
          },
          "SageMakerMachineLearningModelResourceData": {
            "SageMakerJobArn": "string",
            "DestinationPath": "string"
          }
        }
      }
    ]
  }
}
```
GetResourceDefinitionVersionResponse

Information about a resource definition version.

type: object

Arn

Arn of the resource definition version.

type: string

Id

The ID of the resource definition version.

type: string

Version

The version of the resource definition version.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the resource definition version was created.

type: string

Definition

Information about a resource definition version.

type: object

Resources

A list of resources.

type: array

items: Resource (p. 508)

Resource

Information about a resource.

type: object

required: ["Id", "Name", "ResourceDataContainer"]

Id

The resource ID, used to refer to a resource in the Lambda function configuration. Max length is 128 characters with pattern [a-zA-Z0-9:_-]+. This must be unique within a Greengrass group.
type: string

Name

The descriptive resource name, which is displayed on the AWS Greengrass console. Max length 128 characters with pattern [a-zA-Z0-9:_-]+. This must be unique within a Greengrass group.

type: string

ResourceDataContainer

A container for resource data. The container takes only one of the following supported resource data types: LocalDeviceResourceData, LocalVolumeResourceData, SageMakerMachineLearningModelResourceData, S3MachineLearningModelResourceData.

type: object

NOTE: oneOf

LocalDeviceResourceData

Attributes that define a local device resource.

type: object

SourcePath

The local absolute path of the device resource. The source path for a device resource can refer only to a character device or block device under /dev.

type: string

GroupOwnerSetting

Group owner related settings for local resources.

type: object

AutoAddGroupOwner

If true, AWS Greengrass automatically adds the specified Linux OS group owner of the resource to the Lambda process privileges. Thus the Lambda process will have the file access permissions of the added Linux group.

type: boolean

GroupOwner

The name of the Linux OS group whose privileges will be added to the Lambda process. This field is optional.

type: string

LocalVolumeResourceData

Attributes that define a local volume resource.

type: object

SourcePath

The local absolute path of the volume resource on the host. The source path for a volume resource type cannot start with /proc or /sys.

type: string
DestinationPath

The absolute local path of the resource inside the Lambda environment.

type: string

GroupOwnerSetting

Group owner related settings for local resources.

type: object

AutoAddGroupOwner

If true, AWS Greengrass automatically adds the specified Linux OS group owner of the resource to the Lambda process privileges. Thus the Lambda process will have the file access permissions of the added Linux group.

type: boolean

GroupOwner

The name of the Linux OS group whose privileges will be added to the Lambda process. This field is optional.

   type: string

SageMakerMachineLearningModelResourceData

Attributes that define an Amazon SageMaker machine learning resource.

type: object

SageMakerJobArn

The ARN of the Amazon SageMaker training job that represents the source model.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.

type: string

S3MachineLearningModelResourceData

Attributes that define an Amazon S3 machine learning resource.

type: object

S3Uri

The URI of the source model in an Amazon S3 bucket. The model package must be in tar.gz or .zip format.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.

type: string

400

invalid request
GeneralError (p. 476)

```json
{
    "Message": "string",
    "ErrorDetails": [
        {
            "DetailedErrorCode": "string",
            "DetailedErrorMessage": "string"
        }
    ]
}
```

GeneralError

General error information.

- type: object
- required: ["Message"]

Message

A message containing information about the error.

- type: string

ErrorDetails

A list of error details.

- type: array
- items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

- type: object

DetailedErrorCode

A detailed error code.

- type: string

DetailedErrorMessage

A detailed error message.

- type: string

/greengrass/definition/subscriptions

GET

GET /greengrass/definition/subscriptions

Operation ID: ListSubscriptionDefinitions (p. 187)

Retrieves a list of subscription definitions.

Produces: application/json
Query Parameters

MaxResults (p. 450)

The maximum number of results to be returned per request.

where used: query; required: false

type: integer

NextToken (p. 451)

The token for the next set of results, or null if there are no additional results.

where used: query; required: false

type: string

CLI

```bash
aws greengrass list-subscription-definitions
  [--max-results <value>] \
  [--next-token <value>] \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "MaxResults": "integer",
  "NextToken": "string"
}
```

Responses

200 (ListSubscriptionDefinitionsResponse)

ListDefinitionsResponse (p. 498)

```json
{
  "Definitions": [
    {
      "Name": "string",
      "Id": "string",
      "Arn": "string",
      "LastUpdatedTimestamp": "string",
      "CreationTimestamp": "string",
      "LatestVersion": "string",
      "LatestVersionArn": "string"
    }
  ],
  "NextToken": "string"
}
```

ListDefinitionsResponse

A list of definitions.

type: object
Definitions

Information about a definition.

type: array

items: DefinitionInformation (p. 463)

DefinitionInformation

Information about a definition.

type: object

Name

The name of the definition.

type: string

Id

The ID of the definition.

type: string

Arn

The ARN of the definition.

type: string

LastUpdatedTimestamp

The time, in milliseconds since the epoch, when the definition was last updated.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the definition was created.

type: string

LatestVersion

The latest version of the definition.

type: string

LatestVersionArn

The ARN of the latest version of the definition.

type: string

NextToken

The token for the next set of results, or null if there are no additional results.

type: string

POST

POST /greengrass/definition/subscriptions

Operation ID: CreateSubscriptionDefinition (p. 68)
Creates a subscription definition. You may provide the initial version of the subscription definition now or use `CreateSubscriptionDefinitionVersion` at a later time.

Produces: application/json

**Body Parameters**

*CreateSubscriptionDefinitionRequestBody (p. 446)*

where used: body; required: true

```
{  
    "Name": "string",
    "InitialVersion": {  
        "Subscriptions": [  
            {  
                "Id": "string",
                "Source": "string",
                "Subject": "string",
                "Target": "string"
            }
        ]  
    }
}
```

**Name**

The name of the subscription definition.

  type: string

**InitialVersion**

Information about a subscription definition version.

  type: object

**Subscriptions**

A list of subscriptions.

  type: array

  items: **Subscription (p. 518)**

**Subscription**

Information about a subscription.

  type: object

  required: ["Id", "Source", "Subject", "Target"]

**Id**

The id of the subscription.

  type: string

**Source**

The source of the subscription. Can be a thing ARN, a Lambda function ARN, 'cloud' (which represents the AWS IoT cloud), or 'GGShadowService'.

  type: string
Subject

The subject of the message.

type: string

Target

Where the message is sent to. Can be a thing ARN, a Lambda function ARN, 'cloud' (which represents the AWS IoT cloud), or 'GGShadowService'.

type: string

Header Parameters

**X-Amzn-Client-Token (p. 455)**

A client token used to correlate requests and responses.

where used: header; required: false

type: string

CLI

```bash
aws greengrass create-subscription-definition \
  [--name <value>] \
  [--initial-version <value>] \
  [--amzn-client-token <value>] \ 
  [--cli-input-json <value>] \ 
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "Name": "string",
  "InitialVersion": {
    "Subscriptions": [
      {
        "Id": "string",
        "Source": "string",
        "Subject": "string",
        "Target": "string"
      }
    ],
  },
  "AmznClientToken": "string"
}
```

Responses

**200 (CreateSubscriptionDefinitionResponse)**

**DefinitionInformation (p. 463)**

```json
{
  "Name": "string",
  "Id": "string",
  "Arn": "string",
  "LastUpdatedTimestamp": "string",
  "Subscriptions": [
    {
      "Id": "string",
      "Source": "string",
      "Subject": "string",
      "Target": "string"
    }
  ],
  "AmznClientToken": "string"
}
```
"CreationTimestamp": "string",
"LatestVersion": "string",
"LatestVersionArn": "string"
}

DefinitionInformation

Information about a definition.

type: object

Name

The name of the definition.

type: string

Id

The ID of the definition.

type: string

Arn

The ARN of the definition.

type: string

LastUpdatedTimestamp

The time, in milliseconds since the epoch, when the definition was last updated.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the definition was created.

type: string

LatestVersion

The latest version of the definition.

type: string

LatestVersionArn

The ARN of the latest version of the definition.

type: string

400

invalid request

GeneralError (p. 476)
GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

/getgreengrass/definition/subscriptions/SubscriptionDefinitionId

GET

GET /getgreengrass/definition/subscriptions/SubscriptioฑDefinitionId

Operation ID: GetSubscriptionDefinition (p. 144)

Retrieves information about a subscription definition.

Produces: application/json

Path Parameters

SubscriptionDefinitionId (p. 452)

The ID of the subscription definition.

where used: path; required: true


type: string
CLI

```bash
aws greengrass get-subscription-definition \
  --subscription-definition-id <value> \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "SubscriptionDefinitionId": "string"
}
```

Responses

200 (GetSubscriptionDefinitionResponse)

**DefinitionInformation (p. 463)**

```json
{
  "Name": "string",
  "Id": "string",
  "Arn": "string",
  "LastUpdatedTimestamp": "string",
  "CreationTimestamp": "string",
  "LatestVersion": "string",
  "LatestVersionArn": "string"
}
```

**DefinitionInformation**

Information about a definition.

type: object

**Name**

The name of the definition.

type: string

**Id**

The ID of the definition.

type: string

**Arn**

The ARN of the definition.

type: string

**LastUpdatedTimestamp**

The time, in milliseconds since the epoch, when the definition was last updated.

type: string

**CreationTimestamp**

The time, in milliseconds since the epoch, when the definition was created.

type: string
LatestVersion

The latest version of the definition.

type: string

LatestVersionArn

The ARN of the latest version of the definition.

type: string

400

invalid request

GeneralError (p. 476)

```
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string
DELETE

DELETE /greengrass/definition/subscriptions/SubscriptionDefinitionId

Operation ID: DeleteSubscriptionDefinition (p. 86)

Deletes a subscription definition.

Produces: application/json

Path Parameters

SubscriptionDefinitionId (p. 452)

The ID of the subscription definition.

where used: path; required: true

type: string

CLI

aws greengrass delete-subscription-definition \
  --subscription-definition-id <value> \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]

cli-input-json format:

{
  "SubscriptionDefinitionId": "string"
}

Responses

200

success

Empty (p. 467)

{

}

Empty Schema

Empty

type: object

400

invalid request

GeneralError (p. 476)

{

}
"Message": "string",
"ErrorDetails": [
  {
    "DetailedErrorCode": "string",
    "DetailedErrorMessage": "string"
  }
]
}

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

PUT

PUT /greengrass/definition/subscriptions/SubscriptionDefinitionId

Operation ID: UpdateSubscriptionDefinition (p. 210)

Updates a subscription definition.

Produces: application/json

Body Parameters

UpdateDefinitionRequestBody (p. 454)

where used: body; required: true
Name

The name of the definition.

required: true
type: string

Path Parameters

SubscriptionDefinitionId (p. 452)

The ID of the subscription definition.

where used: path; required: true
type: string

CLI

```
aws greengrass update-subscription-definition \
  --subscription-definition-id <value> \
  [--name <value>] \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```
{
  "SubscriptionDefinitionId": "string",
  "Name": "string"
}
```

Responses

200

success

Empty (p. 467)

```
{
}
```

400

invalid request
AWS Greengrass API Reference
/greengrass/definition/subscriptions/
SubscriptionDefinitionId/versions

GeneralError (p. 476)

```json
{
   "Message": "string",
   "ErrorDetails": [
   {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
   }
   ]
}
```

GeneralError
General error information.
type: object
required: ["Message"]

Message
A message containing information about the error.
type: string

ErrorDetails
A list of error details.
type: array
items: ErrorDetail (p. 468)

ErrorDetail
Details about the error.
type: object

DetailedErrorCode
A detailed error code.
type: string

DetailedErrorMessage
A detailed error message.
type: string

/regreengrass/definition/subscriptions/
SubscriptionDefinitionId/versions

POST

POST /greengrass/definition/subscriptions/Subscriptio
SubscriptionDefinitionId/versions

Operation ID: CreateSubscriptionDefinitionVersion (p. 72)

Creates a version of a subscription definition which has already been defined.
Produce: application/json

**Body Parameters**

*CreateSubscriptionDefinitionVersionRequestBody (p. 447)*

where used: body; required: true

```json
{
    "Subscriptions": [
        {
            "Id": "string",
            "Source": "string",
            "Subject": "string",
            "Target": "string"
        }
    ]
}
```

**SubscriptionDefinitionVersion**

Information about a subscription definition version.

type: object

**Subscriptions**

A list of subscriptions.

type: array

items: **Subscription (p. 518)**

**Subscription**

Information about a subscription.

type: object

required: ["Id", "Source", "Subject", "Target"]

**Id**

The id of the subscription.

type: string

**Source**

The source of the subscription. Can be a thing ARN, a Lambda function ARN, 'cloud' (which represents the AWS IoT cloud), or 'GGShadowService'.

type: string

**Subject**

The subject of the message.

type: string

**Target**

Where the message is sent to. Can be a thing ARN, a Lambda function ARN, 'cloud' (which represents the AWS IoT cloud), or 'GGShadowService'.  


type: string

Header Parameters

X-Amzn-Client-Token (p. 455)

A client token used to correlate requests and responses.

where used: header; required: false

type: string

Path Parameters

SubscriptionDefinitionId (p. 452)

The ID of the subscription definition.

where used: path; required: true

type: string

CLI

```
aws greengrass create-subscription-definition-version \
  --subscription-definition-id <value> \ 
  [--subscriptions <value>] \ 
  [--amzn-client-token <value>] \ 
  [--cli-input-json <value>] \ 
  [--generate-cli-skeleton]
```

cli-input-json format:

```
{
  "SubscriptionDefinitionId": "string",
  "Subscriptions": [
    {
      "Id": "string",
      "Source": "string",
      "Subject": "string",
      "Target": "string"
    },
  ],
  "AmznClientToken": "string"
}
```

Responses

200 (CreateSubscriptionDefinitionVersionResponse)

```
VersionInformation (p. 521)
```

```
{
  "Arn": "string",
  "Id": "string",
  "Version": "string",
  "CreationTimestamp": "string"
}
```
VersionInformation

Information about a version.

type: object

Arn

The ARN of the version.

type: string

Id

The ID of the version.

type: string

Version

The unique ID of the version.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the version was created.

type: string

400

invalid request

GeneralError (p. 476)

{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array
items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

GET

GET /greengrass/definition/subscriptions/SubscriptionDefinitionId/versions

Operation ID: ListSubscriptionDefinitionVersions (p. 184)

Lists the versions of a subscription definition.

Produces: application/json

Path Parameters

SubscriptionDefinitionId (p. 452)

The ID of the subscription definition.

where used: path; required: true

type: string

Query Parameters

NextToken (p. 451)

The token for the next set of results, or null if there are no additional results.

where used: query; required: false

type: string

MaxResults (p. 450)

The maximum number of results to be returned per request.

where used: query; required: false

type: integer

CLI

aws greengrass list-subscription-definition-versions \
--subscription-definition-id <value> \
[--next-token <value>] \
[--max-results <value>] \
[--cli-input-json <value>] \
[--generate-cli-skeleton]

cli-input-json format:

{
   "SubscriptionDefinitionId": "string",
   "NextToken": "string",
   "MaxResults": "integer"
}

Responses

200 (ListSubscriptionDefinitionVersionsResponse)

ListVersionsResponse (p. 502)

{
   "Versions": [
   {
      "Arn": "string",
      "Id": "string",
      "Version": "string",
      "CreationTimestamp": "string"
   }
   ],
   "NextToken": "string"
}

ListVersionsResponse

A list of versions.

   type: object

Versions

   Information about a version.

   type: array

   items: VersionInformation (p. 521)

VersionInformation

   Information about a version.

   type: object

Arn

   The ARN of the version.

   type: string

Id

   The ID of the version.

   type: string
Version

The unique ID of the version.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the version was created.

type: string

NextToken

The token for the next set of results, or null if there are no additional results.

type: string

400

invalid request

**GeneralError (p. 476)**

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

**GeneralError**

General error information.

type: object

required: ["Message"]

**Message**

A message containing information about the error.

type: string

**ErrorDetails**

A list of error details.

type: array

items: **ErrorDetail (p. 468)**

**ErrorDetail**

Details about the error.

type: object

**DetailedErrorCode**

A detailed error code.
type: string

DetailedErrorMessage
A detailed error message.
type: string

/greengrass/definition/subscriptions/
SubscriptionDefinitionId/versions/
SubscriptionDefinitionVersionId

GET

GET /greengrass/definition/subscriptions/SubcriptionDefinitionId/
versions/SubcriptionDefinitionVersionId

Operation ID: GetSubscriptionDefinitionVersion (p. 146)
Retrieves information about a subscription definition version.

Produces: application/json

Path Parameters

SubscriptionDefinitionId (p. 452)
The ID of the subscription definition.
where used: path; required: true
type: string

SubscriptionDefinitionVersionId (p. 452)
The ID of the subscription definition version.
where used: path; required: true
type: string

CLI

aws greengrass get-subscription-definition-version \
   --subscription-definition-id <value> \
   --subscription-definition-version-id <value> \
   [--cli-input-json <value>] \
   [--generate-cli-skeleton]

cli-input-json format:

{  "SubscriptionDefinitionId": "string",
   "SubscriptionDefinitionVersionId": "string"
}
Responses

200 (GetSubscriptionDefinitionVersionResponse)

GetSubscriptionDefinitionVersionResponse (p. 492)

```json
{
  "Arn": "string",
  "Id": "string",
  "Version": "string",
  "CreationTimestamp": "string",
  "Definition": {
    "Subscriptions": [
      { "Id": "string",
        "Source": "string",
        "Subject": "string",
        "Target": "string"
      }
    ]
  }
}
```

GetSubscriptionDefinitionVersionResponse

Information about a subscription definition version.

type: object

Arn

The ARN of the subscription definition version.

type: string

Id

The ID of the subscription definition version.

type: string

Version

The version of the subscription definition version.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the subscription definition version was created.

type: string

Definition

Information about a subscription definition version.

type: object

Subscriptions

A list of subscriptions.

type: array

items: Subscription (p. 518)
Subscription

Information about a subscription.

type: object

required: ["Id", "Source", "Subject", "Target"]

Id

The id of the subscription.

type: string

Source

The source of the subscription. Can be a thing ARN, a Lambda function ARN, 'cloud' (which represents the AWS IoT cloud), or 'GGShadowService'.

type: string

Subject

The subject of the message.

type: string

Target

Where the message is sent to. Can be a thing ARN, a Lambda function ARN, 'cloud' (which represents the AWS IoT cloud), or 'GGShadowService'.

type: string

400

invalid request

GeneralError (p. 476)

{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.
type: array
items: ErrorDetail (p. 468)

ErrorDetail
Details about the error.
type: object

DetailedErrorCode
A detailed error code.
type: string

DetailedErrorMessage
A detailed error message.
type: string

/greengrass/groups

GET
GET /greengrass/groups

Operation ID: ListGroups (p. 172)
Retrieves a list of groups.
Produce: application/json

Query Parameters

MaxResults (p. 450)
The maximum number of results to be returned per request.
where used: query; required: false
type: integer

NextToken (p. 451)
The token for the next set of results, or null if there are no additional results.
where used: query; required: false
type: string

CLI

aws greengrass list-groups \ 
[--max-results <value>] \ 
[--next-token <value>] \ 
[--cli-input-json <value>] \ 
[--generate-cli-skeleton]

cli-input-json format:
{  "MaxResults": "integer",  "NextToken": "string"}

Responses

200 (ListGroupsResponse)

ListGroupsResponse (p. 501)

{  "Groups": [    {      "Name": "string",      "Id": "string",      "Arn": "string",      "LastUpdatedTimestamp": "string",      "CreationTimestamp": "string",      "LatestVersion": "string",      "LatestVersionArn": "string"    }  ],  "NextToken": "string"}

ListGroupsResponse
  type: object
Groups
    Information about a group.
    type: array
    items: GroupInformation (p. 495)
GroupInformation
  Information about a group.
  type: object
Name
  The name of the group.
  type: string
Id
  The ID of the group.
  type: string
Arn
  The ARN of the group.
  type: string
LastUpdatedTimestamp
  The time, in milliseconds since the epoch, when the group was last updated.
type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the group was created.

type: string

LatestVersion

The latest version of the group.

type: string

LatestVersionArn

The ARN of the latest version of the group.

type: string

NextToken

The token for the next set of results, or null if there are no additional results.

type: string

POST

POST /greengrass/groups

Operation ID: CreateGroup (p. 36)

Creates a group. You may provide the initial version of the group or use CreateGroupVersion at a later time.

Produces: application/json

Body Parameters

CreateGroupRequestBody (p. 438)

where used: body; required: true

{
  "Name": "string",
  "InitialVersion": {
    "CoreDefinitionVersionArn": "string",
    "DeviceDefinitionVersionArn": "string",
    "FunctionDefinitionVersionArn": "string",
    "SubscriptionDefinitionVersionArn": "string",
    "LoggerDefinitionVersionArn": "string",
    "ResourceDefinitionVersionArn": "string"
  }
}

Name

The name of the group.

type: string

InitialVersion

Information about a group version.
type: object
CoreDefinitionVersionArn

The ARN of the core definition version for this group.

type: string
DeviceDefinitionVersionArn

The ARN of the device definition version for this group.

type: string
FunctionDefinitionVersionArn

The ARN of the function definition version for this group.

type: string
SubscriptionDefinitionVersionArn

The ARN of the subscription definition version for this group.

type: string
LoggerDefinitionVersionArn

The ARN of the logger definition version for this group.

type: string
ResourceDefinitionVersionArn

The resource definition version ARN for this group.

type: string

Header Parameters

X-Amzn-Client-Token (p. 455)

A client token used to correlate requests and responses.

where used: header; required: false

type: string

CLI

```bash
aws greengrass create-group \
  [--name <value>] \
  [--initial-version <value>] \
  [--amzn-client-token <value>] \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
    "Name": "string",
```
"InitialVersion": {
    "CoreDefinitionVersionArn": "string",
    "DeviceDefinitionVersionArn": "string",
    "FunctionDefinitionVersionArn": "string",
    "SubscriptionDefinitionVersionArn": "string",
    "LoggerDefinitionVersionArn": "string",
    "ResourceDefinitionVersionArn": "string"
},
"AmznClientToken": "string"
}

**Responses**

**200 (CreateGroupResponse)**

Success. The group was created.

**DefinitionInformation (p. 463)**

```
{
    "Name": "string",
    "Id": "string",
    "Arn": "string",
    "LastUpdatedTimestamp": "string",
    "CreationTimestamp": "string",
    "LatestVersion": "string",
    "LatestVersionArn": "string"
}
```

**DefinitionInformation**

Information about a definition.

- **Name**
  - type: string
  - The name of the definition.

- **Id**
  - type: string
  - The ID of the definition.

- **Arn**
  - type: string
  - The ARN of the definition.

- **LastUpdatedTimestamp**
  - type: string
  - The time, in milliseconds since the epoch, when the definition was last updated.

- **CreationTimestamp**
  - type: string
  - The time, in milliseconds since the epoch, when the definition was created.
LatestVersion

   The latest version of the definition.

type: string

LatestVersionArn

   The ARN of the latest version of the definition.

type: string

400

invalid request

GeneralError (p. 476)

{
   "Message": "string",
   "ErrorDetails": [
   {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
   }
   ]
}

GeneralError

   General error information.

type: object

   required: ["Message"]

Message

   A message containing information about the error.

type: string

ErrorDetails

   A list of error details.

type: array

   items: ErrorDetail (p. 468)

ErrorDetail

   Details about the error.

type: object

DetailedErrorCode

   A detailed error code.

type: string

DetailedErrorMessage

   A detailed error message.

type: string
GET /greengrass/groups/GroupId

Operation ID: GetGroup (p. 118)
Retrieves information about a group.
Produces: application/json

Path Parameters

GroupId (p. 449)
The ID of the AWS Greengrass group.
where used: path; required: true
type: string

CLI

```bash
aws greengrass get-group \
  --group-id <value>  \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "GroupId": "string"
}
```

Responses

200 (GetGroupResponse)

  success

  DefinitionInformation (p. 463)

```json
{
  "Name": "string",
  "Id": "string",
  "Arn": "string",
  "LastUpdatedTimestamp": "string",
  "CreationTimestamp": "string",
  "LatestVersion": "string",
  "LatestVersionArn": "string"
}
```

DefinitionInformation

  Information about a definition.
type: object
Name
The name of the definition.
type: string
Id
The ID of the definition.
type: string
Arn
The ARN of the definition.
type: string
LastUpdatedTimestamp
The time, in milliseconds since the epoch, when the definition was last updated.
type: string
CreationTimestamp
The time, in milliseconds since the epoch, when the definition was created.
type: string
LatestVersion
The latest version of the definition.
type: string
LatestVersionArn
The ARN of the latest version of the definition.
type: string

400
invalid request

GeneralError (p. 476)

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

GeneralError
General error information.
type: object
required: ["Message"]
DELETE

DELETE /greengrass/groups/\*GroupId\*

Operation ID: DeleteGroup (p. 80)

Deletes a group.

Produces: application/json

Path Parameters

GroupId (p. 449)

The ID of the AWS Greengrass group.

where used: path; required: true

type: string

CLI

```
aws greengrass delete-group \\n  --group-id <value> \\n  [--cli-input-json <value>] \\n  [--generate-cli-skeleton]
```

cli-input-json format:

```
{
```

"GroupId": "string"
}

**Responses**

**200**

success

*Empty (p. 467)*

```json
{
}
```

*Empty Schema*

- Empty
  - type: object

**400**

invalid request

*GeneralError (p. 476)*

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

*GeneralError*

- General error information.
  - type: object
    - required: ["Message"]
  - Message
    - A message containing information about the error.
      - type: string
  - ErrorDetails
    - A list of error details.
      - type: array
        - items: *ErrorDetail (p. 468)*
  - ErrorDetail
    - Details about the error.
      - type: object

---

364
DetailedErrorCode

A detailed error code.
type: string

DetailedErrorMessage

A detailed error message.
type: string

PUT

PUT /greengrass/groups/{GroupId}

Operation ID: UpdateGroup (p. 201)

Updates a group.

Produces: application/json

Body Parameters

UpdateDefinitionRequestBody (p. 454)

where used: body; required: true

```
{
   "Name": "string"
}
```

Name

The name of the definition.

required: true

type: string

Path Parameters

GroupId (p. 449)

The ID of the AWS Greengrass group.

where used: path; required: true

type: string

CLI

```
aws greengrass update-group
   --group-id <value> \n   [--name <value>] \n   [--cli-input-json <value>] \n   [--generate-cli-skeleton]
```
cli-input-json format:

```json
{
   "GroupId": "string",
   "Name": "string"
}
```

**Responses**

**200**

success

Empty (p. 467)

```json
{
}
```

Empty Schema

Empty
type: object

**400**

invalid request

GeneralError (p. 476)

```json
{
   "Message": "string",
   "ErrorDetails": [
      {
         "DetailedErrorCode": "string",
         "DetailedErrorMessage": "string"
      }
   ]
}
```

GeneralError

General error information.
type: object

required: ["Message"]

Message

A message containing information about the error.
type: string

ErrorDetails

A list of error details.
type: array

items: ErrorDetail (p. 468)
ErrorDetail
  Details about the error.
  type: object
DetailedErrorCode
  A detailed error code.
  type: string
DetailedErrorMessage
  A detailed error message.
  type: string

/greengrass/groups/GroupId/certificateauthorities

GET
GET /greengrass/groups/GroupId/certificateauthorities

Operation ID: ListGroupCertificateAuthorities (p. 167)

Retrieves the current CAs for a group.

Produces: application/json

Path Parameters

GroupId (p. 449)
  The ID of the AWS Greengrass group.
  where used: path; required: true
  type: string

CLI

aws greengrass list-group-certificate-authorities \
  --group-id <value> \ 
  [--cli-input-json <value>] \ 
  [--generate-cli-skeleton]

cli-input-json format:

{
  "GroupId": "string"
}

Responses

200
  Success. The response body contains the PKI Configuration.
ListGroupCertificateAuthoritiesResponse (p. 500)

```json
{
  "GroupCertificateAuthorities": [
    {
      "GroupCertificateAuthorityId": "string",
      "GroupCertificateAuthorityArn": "string"
    }
  ]
}
```

**ListGroupCertificateAuthoritiesResponse**  
**type:** object  

**GroupCertificateAuthorities**  
A list of certificate authorities associated with the group.  
**type:** array  

**items:** [GroupCertificateAuthorityProperties (p. 494)]  

**Group Certificate Authority Properties**  
Information about a certificate authority for a group.  
**type:** object  

**GroupCertificateAuthorityId**  
The ID of the certificate authority for the group.  
**type:** string  

**GroupCertificateAuthorityArn**  
The ARN of the certificate authority for the group.  
**type:** string

400  
invalid request  

**GeneralError (p. 476)**

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

**GeneralError**  
General error information.  
**type:** object  

required: ["Message"]
Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

500

server error

GeneralError (p. 476)

{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array
items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

POST

POST /greengrass/groups/GroupId/certificateauthorities

Operation ID: CreateGroupCertificateAuthority (p. 39)

Creates a CA for the group. If a CA already exists, it will rotate the existing CA.

Produces: application/json

Header Parameters

X-Amzn-Client-Token (p. 455)

A client token used to correlate requests and responses.

where used: header; required: false

type: string

Path Parameters

GroupId (p. 449)

The ID of the AWS Greengrass group.

where used: path; required: true

type: string

CLI

```
aws greengrass create-group-certificate-authority \
  --group-id <value> \
  [--amzn-client-token <value>] \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:
```json
{
   "GroupId": "string",
   "AmznClientToken": "string"
}
```

**Responses**

**200**

Success. The response body contains the new active CA ARN.

*CreateGroupCertificateAuthorityResponse* (p. 461)

```json
{
   "GroupCertificateAuthorityArn": "string"
}
```

*CreateGroupCertificateAuthorityResponse*

- **type**: object
- **GroupCertificateAuthorityArn**
  - The ARN of the group certificate authority.
  - **type**: string

**400**

Invalid request

*GeneralError* (p. 476)

```json
{
   "Message": "string",
   "ErrorDetails": [
      {
         "DetailedErrorCode": "string",
         "DetailedErrorMessage": "string"
      }
   ]
}
```

*GeneralError*

- **General error information.**
  - **type**: object
  - **required**: ["Message"]
- **Message**
  - A message containing information about the error.
  - **type**: string
- **ErrorDetails**
  - A list of error details.
  - **type**: array
items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.
type: object

DetailedErrorCode

A detailed error code.
type: string

DetailedErrorMessage

A detailed error message.
type: string

500

server error

GeneralError (p. 476)

{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}

GeneralError

General error information.
type: object

required: ["Message"]

Message

A message containing information about the error.
type: string

ErrorDetails

A list of error details.
type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.
type: object

DetailedErrorCode

A detailed error code.
/greengrass/groups/GroupId/certificateauthorities/configuration/expiry

GET

GET /greengrass/groups/GroupId/certificateauthorities/configuration/expiry

Operation ID: GetGroupCertificateConfiguration (p. 123)

Retrieves the current configuration for the CA used by the group.

Produces: application/json

Path Parameters

GroupId (p. 449)

The ID of the AWS Greengrass group.

where used: path; required: true

type: string

CLI

aws greengrass get-group-certificate-configuration \   
   --group-id <value> \   
   [--cli-input-json <value>] \   
   [--generate-cli-skeleton]

cli-input-json format:

```json
{
   "GroupId": "string"
}
```

Responses

200

Success. The response body contains the PKI Configuration.

GroupCertificateConfiguration (p. 495)

```json
{
   "GroupId": "string",
   "CertificateAuthorityExpiryInMilliseconds": "string",
```
GroupCertificateConfiguration

Information about a group certificate configuration.

type: object

GroupId

The ID of the group certificate configuration.

type: string

CertificateAuthorityExpiryInMilliseconds

The amount of time remaining before the certificate authority expires, in milliseconds.

type: string

CertificateExpiryInMilliseconds

The amount of time remaining before the certificate expires, in milliseconds.

type: string

400

invalid request

GeneralError (p. 476)

{
    "Message": "string",
    "ErrorDetails": [
        {
            "DetailedErrorCode": "string",
            "DetailedErrorMessage": "string"
        }
    ]
}

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)
ErrorDetail
   Details about the error.
   type: object
DetailedErrorCode
   A detailed error code.
   type: string
DetailedErrorMessage
   A detailed error message.
   type: string

500
server error

GeneralError (p. 476)

{  
   "Message": "string",  
   "ErrorDetails": [  
      {  
         "DetailedErrorCode": "string",  
         "DetailedErrorMessage": "string"  
      }  
   ]
}

GeneralError

   General error information.
   type: object
   required: ["Message"]
Message

   A message containing information about the error.
   type: string
ErrorDetails

   A list of error details.
   type: array
   items: ErrorDetail (p. 468)
ErrorDetail

   Details about the error.
   type: object
DetailedErrorCode

   A detailed error code.
   type: string
DetailedErrorMessage

A detailed error message.

type: string

PUT

PUT /greengrass/groups/GroupId/certificateauthorities/configuration/expiry

Operation ID: UpdateGroupCertificateConfiguration (p. 203)

Updates the Certificate expiry time for a group.

Produces: application/json

Body Parameters

UpdateGroupCertificateConfigurationRequestBody (p. 454)

where used: body; required: true

```
{
    "CertificateExpiryInMilliseconds": "string"
}
```

updateGroupCertificateConfigurationRequestBody

type: object

required: ["CertificateExpiryInMilliseconds"]

CertificateExpiryInMilliseconds

The amount of time remaining before the certificate expires, in milliseconds.

type: string

Path Parameters

GroupId (p. 449)

The ID of the AWS Greengrass group.

where used: path; required: true

type: string

CLI

```
aws greengrass update-group-certificate-configuration \
  --group-id <value> \
  [--certificate-expiry-in-milliseconds <value>] \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:
AWS Greengrass API Reference
/greengrass/groups/GroupId/
certificateauthorities/configuration/expiry

```json
{
   "GroupId": "string",
   "CertificateExpiryInMilliseconds": "string"
}
```

**Responses**

**200**

Success. The response body contains the PKI Configuration.

GroupCertificateConfiguration (p. 495)

```json
{
   "GroupId": "string",
   "CertificateAuthorityExpiryInMilliseconds": "string",
   "CertificateExpiryInMilliseconds": "string"
}
```

GroupCertificateConfiguration

Information about a group certificate configuration.

- **GroupId**
  - type: object
  - GroupId
    - The ID of the group certificate configuration.
    - type: string
  - CertificateAuthorityExpiryInMilliseconds
    - The amount of time remaining before the certificate authority expires, in milliseconds.
    - type: string
  - CertificateExpiryInMilliseconds
    - The amount of time remaining before the certificate expires, in milliseconds.
    - type: string

**400**

invalid request

GeneralError (p. 476)

```json
{
   "Message": "string",
   "ErrorDetails": [
   {  
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
   }
   ]
}
```

GeneralError

General error information.
type: object
required: ["Message"]

Message
A message containing information about the error.
type: string

ErrorDetails
A list of error details.
type: array
items: ErrorDetail (p. 468)

ErrorDetail
Details about the error.
type: object

DetailedErrorCode
A detailed error code.
type: string

DetailedErrorMessage
A detailed error message.
type: string

500
server error

GeneralError (p. 476)

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}

GeneralError
General error information.
type: object
required: ["Message"]

Message
A message containing information about the error.
type: string
ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

/greengrass/groups/GroupId/certificateauthorities/CertificateAuthorityId

GET

GET /greengrass/groups/GroupId/certificateauthorities/CertificateAuthorityId

Operation ID: GetGroupCertificateAuthority (p. 120)

Retreives the CA associated with a group. Returns the public key of the CA.

Produces: application/json

Path Parameters

GroupId (p. 449)

The ID of the AWS Greengrass group.

where used: path; required: true

type: string

CertificateAuthorityId (p. 430)

The ID of the certificate authority.

where used: path; required: true

type: string

CLI

aws greengrass get-group-certificate-authority \
--group-id <value> \n--certificate-authority-id <value> \n[[--cli-input-json <value>] \n[[--generate-cli-skeleton]]

cli-input-json format:

```json
{
  "GroupId": "string",
  "CertificateAuthorityId": "string"
}
```

**Responses**

**200**

Success. The response body contains the PKI Configuration.

GetGroupCertificateAuthorityResponse (p. 485)

```json
{
  "PemEncodedCertificate": "string",
  "GroupCertificateAuthorityArn": "string",
  "GroupCertificateAuthorityId": "string"
}
```

GetGroupCertificateAuthorityResponse

Information about a certificate authority for a group.

- **type**: object
  - **PemEncodedCertificate**
    - The PEM encoded certificate for the group.
    - **type**: string
  - **GroupCertificateAuthorityArn**
    - The ARN of the certificate authority for the group.
    - **type**: string
  - **GroupCertificateAuthorityId**
    - The ID of the certificate authority for the group.
    - **type**: string

**400**

invalid request

GeneralError (p. 476)

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```
GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

500

server error

GeneralError (p. 476)

{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}

GeneralError

General error information.

Type: object

required: ["Message"]
Message
A message containing information about the error.
type: string

ErrorDetails
A list of error details.
type: array
items: ErrorDetail (p. 468)

ErrorDetail
Details about the error.
type: object

DetailedErrorCode
A detailed error code.
type: string

DetailedErrorMessage
A detailed error message.
type: string

/greengrass/groups/GroupId/deployments

POST
POST /greengrass/groups/GroupId/deployments

Operation ID: CreateDeployment (p. 15)

Creates a deployment.

Produces: application/json

Body Parameters

CreateDeploymentRequestBody (p. 432)

where used: body; required: true

```json
{
  "DeploymentType": "NewDeployment|Redeployment|ResetDeployment|ForceResetDeployment",
  "DeploymentId": "string",
  "GroupVersionId": "string"
}
```

CreateDeploymentRequest

Information about a deployment.
type: object
required: ["DeploymentType"]

**DeploymentType**

type: string

enum: ["NewDeployment", "Redeployment", "ResetDeployment", "ForceResetDeployment"]

**DeploymentId**

The ID of the deployment if you wish to redeploy a previous deployment.

type: string

**GroupVersionId**

The ID of the group version to be deployed.

type: string

**Header Parameters**

**X-Amzn-Client-Token** (p. 455)

A client token used to correlate requests and responses.

where used: header; required: false

type: string

**Path Parameters**

**GroupId** (p. 449)

The ID of the AWS Greengrass group.

where used: path; required: true

type: string

**CLI**

```bash
aws greengrass create-deployment \
  --group-id <value> \
  [--deployment-type <value>] \
  [--deployment-id <value>] \
  [--group-version-id <value>] \
  [--amzn-client-token <value>] \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "GroupId": "string",
  "DeploymentType": "NewDeployment|Redeployment|ResetDeployment|ForceResetDeployment",
  "DeploymentId": "string",
  "GroupVersionId": "string",
  "AmznClientToken": "string"
}
```
Responses

200

Success. The group was deployed.

CreateDeploymentResponse (p. 461)

```json
{
  "Deployment Arn": "string",
  "Deployment Id": "string"
}
```

CreateDeploymentResponse

type: object

DeploymentArn

The ARN of the deployment.

type: string

DeploymentId

The ID of the deployment.

type: string

400

invalid request

GeneralError (p. 476)

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array
items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.
type: object

DetailedErrorCode

A detailed error code.
type: string

DetailedErrorMessage

A detailed error message.
type: string

GET

GET /greengrass/groups/GroupId/deployments

Operation ID: ListDeployments (p. 154)

Returns a history of deployments for the group.

Produce: application/json

Path Parameters

GroupId (p. 449)

The ID of the AWS Greengrass group.
where used: path; required: true
type: string

Query Parameters

MaxResults (p. 450)

The maximum number of results to be returned per request.
where used: query; required: false
type: integer

NextToken (p. 451)

The token for the next set of results, or null if there are no additional results.
where used: query; required: false
type: string

CLI

aws greengrass list-deployments \
CLI commands:

```
--group-id <value> \
[--max-results <value>] \
[--next-token <value>] \
[--cli-input-json <value>] \
[--generate-cli-skeleton]
```

cli-input-json format:

```
{
  "GroupId": "string",
  "MaxResults": "integer",
  "NextToken": "string"
}
```

Responses

200

Success. The response body contains the list of deployments for the given group.

**ListDeploymentsResponse (p. 499)**

```
{
  "Deployments": [
    {
      "GroupArn": "string",
      "DeploymentId": "string",
      "DeploymentArn": "string",
      "DeploymentType": "NewDeployment|Redeployment|ResetDeployment|ForceResetDeployment",
      "CreatedAt": "string"
    }
  ],
  "NextToken": "string"
}
```

**ListDeploymentsResponse**

- type: object
- Deployments
  - type: array
  - items: Deployment (p. 464)
- Deployment
  - Information about a deployment.
  - type: object
  - GroupArn
    - The ARN of the group for this deployment.
    - type: string
  - DeploymentId
    - The ID of the deployment.
    - type: string
DeploymentArn

The ARN of the deployment.

type: string

DeploymentType

The type of deployment.

type: string
enum: ["NewDeployment", "Redeployment", "ResetDeployment", "ForceResetDeployment"]

CreatedAt

The time, in milliseconds since the epoch, when the deployment was created.

type: string

NextToken

The token for the next set of results, or null if there are no additional results.
in: query

type: string

400
invalid request

GeneralError (p. 476)

{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}

GeneralError

General error information.

type: object
required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array
items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.
/greengrass/groups/GroupId/deployments/$reset

POST

POST /greengrass/groups/GroupId/deployments/$reset

Operation ID: ResetDeployments (p. 189)

Resets a group's deployments.

Produces: application/json

Body Parameters

ResetDeploymentsRequestBody (p. 451)

Information needed to reset deployments.

where used: body; required: true

```json
{
    "Force": true
}
```

ResetDeploymentsRequest

Information about a group reset request.

type: object

Force

If true, performs a best-effort only core reset.

type: boolean

Header Parameters

X-Amzn-Client-Token (p. 455)

A client token used to correlate requests and responses.

where used: header; required: false

type: string
Path Parameters

GroupId (p. 449)
   The ID of the AWS Greengrass group.
   where used: path; required: true
type: string

CLI

```
aws greengrass reset-deployments
   --group-id <value>
   [--force | --no-force]
   [--amzn-client-token <value>]
   [--cli-input-json <value>]
   [--generate-cli-skeleton]
```

cli-input-json format:

```
{
   "GroupId": "string",
   "Force": "boolean",
   "AmznClientToken": "string"
}
```

Responses

200
   Success. The group's deployments were reset.
   ResetDeploymentsResponse (p. 508)

```
{
   "DeploymentArn": "string",
   "DeploymentId": "string"
}
```

ResetDeploymentsResponse
type: object
DeploymentArn
   The ARN of the deployment.
type: string
DeploymentId
   The ID of the deployment.
type: string

400
   invalid request
   GeneralError (p. 476)
GET

GET /greengrass/groups/GroupId/deployments/DeploymentId/status

Operation ID: GetDeploymentStatus (p. 103)

Returns the status of a deployment.
Path Parameters

**GroupId** (p. 449)

The ID of the AWS Greengrass group.

where used: path; required: true
type: string

**DeploymentId** (p. 448)

The ID of the deployment.

where used: path; required: true
type: string

CLI

```bash
aws greengrass get-deployment-status \ 
  --group-id <value> \ 
  --deployment-id <value> \ 
  [--cli-input-json <value>] \ 
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "GroupId": "string",
  "DeploymentId": "string"
}
```

Responses

**200**

Success. The response body contains the status of the deployment for the group.

GetDeploymentStatusResponse (p. 480)

```json
{
  "DeploymentStatus": "string",
  "DeploymentType": "NewDeployment|Redeployment|ResetDeployment|ForceResetDeployment",
  "UpdatedAt": "string",
  "ErrorMessage": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

GetDeploymentStatusResponse

Information about the status of a deployment for a group.
DeploymentStatus

The status of the deployment.

type: string

DeploymentType

The time, in milliseconds since the epoch, when the deployment status was updated.

type: string

ErrorMessage

Error message

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

400

invalid request

GeneralError (p. 476)

{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
GeneralError
  General error information.
  type: object
  required: ["Message"]
Message
  A message containing information about the error.
  type: string
ErrorDetails
  A list of error details.
  type: array
  items: ErrorDetail (p. 468)
ErrorDetail
  Details about the error.
  type: object
DetailedErrorCode
  A detailed error code.
  type: string
DetailedErrorMessage
  A detailed error message.
  type: string

/greengrass/groups/GroupId/role

PUT

PUT /greengrass/groups/GroupId/role

Operation ID: AssociateRoleToGroup (p. 3)

Associates a role with a group. Your AWS Greengrass core will use the role to access AWS cloud services. The role’s permissions should allow Greengrass core Lambda functions to perform actions against the cloud.

Body Parameters

AssociateRoleToGroupRequestBody (p. 429)

  where used: body; required: true

  ```
  {
    "RoleArn": "string"
  }
  ```
AssociateRoleToGroupRequest
type: object
required: ["RoleArn"]

RoleArn
The ARN of the role you wish to associate with this group.
type: string

Path Parameters

GroupId (p. 449)
The ID of the AWS Greengrass group.
where used: path; required: true
type: string

CLI

```
aws greengrass associate-role-to-group \
  --group-id <value> \
  [--role-arn <value>] \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

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

Responses

200
success

AssociateRoleToGroupResponse (p. 494)

```
{
  "AssociatedAt": "string"
}
```

Group
type: object

AssociatedAt
The time, in milliseconds since the epoch, when the role ARN was associated with the group.
type: string
400

invalid request

GeneralError (p. 476)

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

500

server error

GeneralError (p. 476)

```json
{
  "Message": "string",
  
```
"ErrorDetails": [
    {
        "DetailedErrorCode": "string",
        "DetailedErrorMessage": "string"
    }
]

GeneralError

General error information.

  type: object

  required: ["Message"]

Message

A message containing information about the error.

  type: string

ErrorDetails

A list of error details.

  type: array

  items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

  type: object

DetailedErrorCode

A detailed error code.

  type: string

DetailedErrorMessage

A detailed error message.

  type: string

DELETE

DELETE /greengrass/groups/GroupId/role

Operation ID: DisassociateRoleFromGroup (p. 87)

Disassociates the role from a group.

Path Parameters

GroupId (p. 449)

  The ID of the AWS Greengrass group.

  where used: path; required: true
type: string

**CLI**

```
aws greengrass disassociate-role-from-group \
   --group-id <value> \ 
   [--cli-input-json <value>] \ 
   [--generate-cli-skeleton]
```

cli-input-json format:

```
{
   "GroupId": "string"
}
```

**Responses**

**200**

success

DisassociateRoleFromGroupResponse (p. 467)

```
{
   "DisassociatedAt": "string"
}
```

DisassociateRoleFromGroupResponse

- type: object
- DisassociatedAt
  - The time, in milliseconds since the epoch, when the role was disassociated from the group.
  - type: string

**400**

invalid request

GeneralError (p. 476)

```
{
   "Message": "string",
   "ErrorDetails": [
      {
         "DetailedErrorCode": "string",
         "DetailedErrorMessage": "string"
      }
   ]
}
```

GeneralError

General error information.

- type: object
- required: ["Message"]
Message
A message containing information about the error.
type: string

ErrorDetails
A list of error details.
type: array
items: ErrorDetail (p. 468)

ErrorDetail
Details about the error.
type: object

DetailedErrorCode
A detailed error code.
type: string

DetailedErrorMessage
A detailed error message.
type: string

500
server error

GeneralError (p. 476)

{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}

GeneralError
General error information.
type: object

required: ["Message"]

Message
A message containing information about the error.
type: string

ErrorDetails
A list of error details.
type: array
items: ErrorDetail (p. 468)

ErrorDetail
  Details about the error.
  type: object
  DetailedErrorCode
    A detailed error code.
    type: string
  DetailedErrorMessage
    A detailed error message.
    type: string

GET

GET /greengrass/groups/GroupId/role

Operation ID: GetAssociatedRole (p. 91)

Retrieves the role associated with a particular group.

Path Parameters

GroupId (p. 449)

  The ID of the AWS Greengrass group.
  where used: path; required: true
  type: string

CLI

aws greengrass get-associated-role \
  --group-id <value> \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]

cli-input-json format:

{
  "GroupId": "string"
}

Responses

200

  success
  GetAssociatedRoleResponse (p. 477)
GetAssociatedRoleResponse

type: object

AssociatedAt

The time when the role was associated with the group.

type: string

RoleArn

The ARN of the role that is associated with the group.

type: string

400

invalid request

GeneralError (p. 476)

{  
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object
DetailedErrorCode

A detailed error code.
type: string

DetailedErrorMessage

A detailed error message.
type: string

500

server error

GeneralError (p. 476)

```
{
    "Message": "string",
    "ErrorDetails": [
    {
        "DetailedErrorCode": "string",
        "DetailedErrorMessage": "string"
    }
    ]
}
```

GeneralError

General error information.
type: object
required: ["Message"]

Message

A message containing information about the error.
type: string

ErrorDetails

A list of error details.
type: array
items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.
type: object

DetailedErrorCode

A detailed error code.
type: string

DetailedErrorMessage

A detailed error message.
type: string
/greengrass/groups/GroupId/versions

POST

POST /greengrass/groups/GroupId/versions

Operation ID: CreateGroupVersion (p. 42)

Creates a version of a group which has already been defined.

Produces: application/json

Body Parameters

CreateGroupVersionRequestBody (p. 438)

where used: body; required: true

```json
{
    "CoreDefinitionVersionArn": "string",
    "DeviceDefinitionVersionArn": "string",
    "FunctionDefinitionVersionArn": "string",
    "SubscriptionDefinitionVersionArn": "string",
    "LoggerDefinitionVersionArn": "string",
    "ResourceDefinitionVersionArn": "string"
}
```

GroupVersion

Information about a group version.

type: object

CoreDefinitionVersionArn

The ARN of the core definition version for this group.

type: string

DeviceDefinitionVersionArn

The ARN of the device definition version for this group.

type: string

FunctionDefinitionVersionArn

The ARN of the function definition version for this group.

type: string

SubscriptionDefinitionVersionArn

The ARN of the subscription definition version for this group.

type: string

LoggerDefinitionVersionArn

The ARN of the logger definition version for this group.

type: string
ResourceDefinitionVersionArn

The resource definition version ARN for this group.

type: string

Header Parameters

X-Amzn-Client-Token (p. 455)

A client token used to correlate requests and responses.

where used: header; required: false

type: string

Path Parameters

GroupId (p. 449)

The ID of the AWS Greengrass group.

where used: path; required: true

type: string

CLI

```bash
aws greengrass create-group-version \
  --group-id <value> \
  [--core-definition-version-arn <value>] \
  [--device-definition-version-arn <value>] \
  [--function-definition-version-arn <value>] \
  [--subscription-definition-version-arn <value>] \
  [--logger-definition-version-arn <value>] \
  [--resource-definition-version-arn <value>] \
  [--amzn-client-token <value>] \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "GroupId": "string",
  "CoreDefinitionVersionArn": "string",
  "DeviceDefinitionVersionArn": "string",
  "FunctionDefinitionVersionArn": "string",
  "SubscriptionDefinitionVersionArn": "string",
  "LoggerDefinitionVersionArn": "string",
  "ResourceDefinitionVersionArn": "string",
  "AmznClientToken": "string"
}
```

Responses

200 (CreateGroupVersionResponse)

Success. The response contains information about the group version.
**VersionInformation (p. 521)**

```json
{
    "Arn": "string",
    "Id": "string",
    "Version": "string",
    "CreationTimestamp": "string"
}
```

**VersionInformation**

Information about a version.

type: object

**Arn**

The ARN of the version.

type: string

**Id**

The ID of the version.

type: string

**Version**

The unique ID of the version.

type: string

**CreationTimestamp**

The time, in milliseconds since the epoch, when the version was created.

type: string

400

invalid request

**GeneralError (p. 476)**

```json
{
    "Message": "string",
    "ErrorDetails": [
    {
        "DetailedErrorCode": "string",
        "DetailedErrorMessage": "string"
    }
    ]
}
```

**GeneralError**

General error information.

type: object

required: ["Message"]

**Message**

A message containing information about the error.

404
GET

GET /greengrass/groups/GroupId/versions

Operation ID: ListGroupVersions (p. 170)
Lists the versions of a group.
Produces: application/json

Path Parameters

GroupId (p. 449)

The ID of the AWS Greengrass group.
where used: path; required: true
type: string

Query Parameters

NextToken (p. 451)

The token for the next set of results, or null if there are no additional results.
where used: query; required: false
type: string

MaxResults (p. 450)

The maximum number of results to be returned per request.
where used: query; required: false
type: integer

CLI

```bash
aws greengrass list-group-versions \
   --group-id <value> \
   [--next-token <value>] \
   [--max-results <value>] \
   [--cli-input-json <value>] \
   [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
   "GroupId": "string",
   "NextToken": "string",
   "MaxResults": "integer"
}
```

Responses

**200 (ListGroupVersionsResponse)**

Success. The response contains the list of versions and metadata for the given group.

**ListVersionsResponse (p. 502)**

```json
{
   "Versions": [
      {
         "Arn": "string",
         "Id": "string",
         "Version": "string",
         "CreationTimestamp": "string"
      },
      "NextToken": "string"
   }
}
```

**ListVersionsResponse**

A list of versions.

type: object

**Versions**

Information about a version.

type: array

items: **VersionInformation (p. 521)**

**VersionInformation**

Information about a version.

type: object

**Arn**

The ARN of the version.
type: string
Id

The ID of the version.

type: string
Version

The unique ID of the version.

type: string
CreationTimestamp

The time, in milliseconds since the epoch, when the version was created.

type: string
NextToken

The token for the next set of results, or null if there are no additional results.

type: string

400

invalid request

GeneralError (p. 476)

{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.
type: object
DetailedErrorCode
  A detailed error code.
  type: string
DetailedErrorMessage
  A detailed error message.
  type: string

/greengrass/groups/GroupId/versions/GroupVersionId

GET

GET /greengrass/groups/GroupId/versions/GroupVersionId

Operation ID: GetGroupVersion (p. 126)

Retrieves information about a group version.

Produces: application/json

Path Parameters

GroupId (p. 449)

  The ID of the AWS Greengrass group.
  where used: path; required: true
  type: string

GroupVersionId (p. 450)

  The ID of the group version.
  where used: path; required: true
  type: string

CLI

aws greengrass get-group-version \
  --group-id <value> \
  --group-version-id <value> \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]

cli-input-json format:

{
  "GroupId": "string",
  "GroupVersionId": "string"
Responses

200

success

GetGroupVersionResponse (p. 486)

{
  "Arn": "string",
  "Id": "string",
  "Version": "string",
  "CreationTimestamp": "string",
  "Definition": {
    "CoreDefinitionVersionArn": "string",
    "DeviceDefinitionVersionArn": "string",
    "FunctionDefinitionVersionArn": "string",
    "SubscriptionDefinitionVersionArn": "string",
    "LoggerDefinitionVersionArn": "string",
    "ResourceDefinitionVersionArn": "string"
  }
}

GetGroupVersionResponse

Information about a group version.
type: object

Arn

The ARN of the group version.
type: string

Id

The ID of the group version.
type: string

Version

The unique ID for the version of the group.
type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the group version was created.
type: string

Definition

Information about a group version.
type: object

CoreDefinitionVersionArn

The ARN of the core definition version for this group.
type: string
DeviceDefinitionVersionArn

The ARN of the device definition version for this group.

type: string

FunctionDefinitionVersionArn

The ARN of the function definition version for this group.

  type: string

SubscriptionDefinitionVersionArn

The ARN of the subscription definition version for this group.

  type: string

LoggerDefinitionVersionArn

The ARN of the logger definition version for this group.

  type: string

ResourceDefinitionVersionArn

The resource definition version ARN for this group.

  type: string

400

invalid request

GeneralError (p. 476)

{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}

GeneralError

General error information.

type: object

  required: ["Message"]

Message

A message containing information about the error.

  type: string

ErrorDetails

A list of error details.

  type: array

  items: ErrorDetail (p. 468)
ErrorDetail
   Details about the error.
   type: object
DetailedErrorCode
   A detailed error code.
   type: string
DetailedErrorMessage
   A detailed error message.
   type: string

/greengrass/servicerole

PUT

PUT /greengrass/servicerole

Operation ID: AssociateServiceRoleToAccount (p. 6)

Associates a role with your account. AWS Greengrass will use the role to access your Lambda functions and AWS IoT resources. This is necessary for deployments to succeed. The role must have at least minimum permissions in the policy AWSGreengrassResourceAccessRolePolicy.

Body Parameters

AssociateServiceRoleToAccountRequestBody (p. 429)

where used: body; required: true

{
   "RoleArn": "string"
}

AssociateServiceRoleToAccountRequest

  type: object
  required: ["RoleArn"]
RoleArn

  The ARN of the service role you wish to associate with your account.
  type: string

CLI

```
aws greengrass associate-service-role-to-account
   
   [--role-arn <value>] 
   
   [--cli-input-json <value>] 
   
   [--generate-cli-skeleton]
```
cli-input-json format:
```json
{
  "RoleArn": "string"
}
```

**Responses**

**200**

success

`AssociateServiceRoleToAccountResponse (p. 458)`
```json
{
  "AssociatedAt": "string"
}
```

`AssociateServiceRoleToAccountResponse`
- type: object

`AssociatedAt`
- The time when the service role was associated with the account.
- type: string

**400**

invalid request

`GeneralError (p. 476)`
```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

`GeneralError`
- General error information.
- type: object
- required: ["Message"]

`Message`
- A message containing information about the error.
- type: string

`ErrorDetails`
- A list of error details.
- type: array
items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

500

server error

GeneralError (p. 476)

{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.
DELETE

DELETE /greengrass/servicerole

Operation ID: DisassociateServiceRoleFromAccount (p. 90)

Disassociates the service role from your account. Without a service role, deployments will not work.

CLI

```bash
aws greengrass disassociate-service-role-from-account  \
[--cli-input-json <value>]  \
[--generate-cli-skeleton]
```

Responses

200

success

DisassociateServiceRoleFromAccountResponse (p. 467)

```json
{
  "DisassociatedAt": "string"
}
```

DisassociateServiceRoleFromAccountResponse

type: object

DisassociatedAt

The time when the service role was disassociated from the account.

type: string

500

server error

GeneralError (p. 476)

```json
{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}
```

GeneralError
   General error information.
   type: object
   required: ["Message"]
Message
   A message containing information about the error.
   type: string
ErrorDetails
   A list of error details.
   type: array
   items: ErrorDetail (p. 468)
ErrorDetail
   Details about the error.
   type: object
DetailedErrorCode
   A detailed error code.
   type: string
DetailedErrorMessage
   A detailed error message.
   type: string

GET
GET /greengrass/servicerole
Operation ID: GetServiceRoleForAccount (p. 142)
Retrieves the service role that is attached to your account.

CLI

aws greengrass get-service-role-for-account \
[--cli-input-json <value>] \
[--generate-cli-skeleton]

Responses

200
success

GetServiceRoleForAccountResponse (p. 492)


"AssociatedAt": "string",
"RoleArn": "string"
}

GetServiceRoleForAccountResponse
type: object

AssociatedAt

The time when the service role was associated with the account.
type: string

RoleArn

The ARN of the role which is associated with the account.
type: string

500

server error

GeneralError (p. 476)

{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}

GeneralError

General error information.
type: object

required: ["Message"]

Message

A message containing information about the error.
type: string

ErrorDetails

A list of error details.
type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.
type: object

DetailedErrorCode

A detailed error code.
/greengrass/things/ThingName/connectivityInfo

GET

GET /greengrass/things/ThingName/connectivityInfo

Operation ID: GetConnectivityInfo (p. 94)

Retrieves the connectivity information for a core.

Produces: application/json

Path Parameters

ThingName (p. 453)

The thing name.

where used: path; required: true

type: string

CLI

```bash
aws greengrass get-connectivity-info \
  --thing-name <value> \
  [--cli-input-json <value>] \
  [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
  "ThingName": "string"
}
```

Responses

200

success

GetConnectivityInfoResponse (p. 477)

```json
{
  "message": "string",
  "ConnectivityInfo": [
  {  
    "Id": "string",
    "HostAddress": "string",
```
"PortNumber": 0x01,
"Metadata": "string"
}
]

Information about a Greengrass core's connectivity.
message

A message about the connectivity info request.
type: string

ConnectivityInfo

Connectivity info list.
type: array
items: ConnectivityInfo (p. 458)

Information about a Greengrass core's connectivity.
required: ["Id", "HostAddress"]

Id

The ID of the connectivity information.
type: string

HostAddress

The endpoint for the Greengrass core. Can be an IP address or DNS.
type: string

PortNumber

The port of the Greengrass core. Usually 8883.
type: integer
format: int32

Metadata

Metadata for this endpoint.
type: string

400

invalid request

GeneralError (p. 476)

{
"Message": "string",
"ErrorDetails": [
{
"DetailedErrorCode": "string",
"DetailedErrorMessage": "string"
}
]
}
GeneralError

General error information.

```json
{ "Message": "string",
 "ErrorDetails": [ 
 { 
 "DetailedErrorCode": "string",
 "DetailedErrorMessage": "string"
 } ]
}
```

GeneralError

General error information.

```json
{ "Message": "string",
 "ErrorDetails": [ 
 { 
 "DetailedErrorCode": "string",
 "DetailedErrorMessage": "string"
 } ]
}
```

500

server error

GeneralError (p. 476)
PUT

PUT /greengrass/things/ThingName/connectivityInfo

Operation ID: UpdateConnectivityInfo (p. 191)

Updates the connectivity information for the core. Any devices that belong to the group which has this core will receive this information in order to find the location of the core and connect to it.

Produces: application/json

Body Parameters

UpdateConnectivityInfoRequestBody (p. 453)

The connectivity information.

where used: body; required:

```json
{
  "ConnectivityInfo": [
    {
      "Id": "string",
      "HostAddress": "string",
      "PortNumber": 0x01,
      "Metadata": "string"
    }
  ]
}
```

Information required to update a Greengrass core's connectivity.
ConnectivityInfo

A list of connectivity info.

type: array

items: ConnectivityInfo (p. 458)

Information about a Greengrass core's connectivity.

required: ["Id", "HostAddress"]

Id

The ID of the connectivity information.

type: string

HostAddress

The endpoint for the Greengrass core. Can be an IP address or DNS.

type: string

PortNumber

The port of the Greengrass core. Usually 8883.

type: integer

format: int32

Metadata

Metadata for this endpoint.

type: string

Path Parameters

**ThingName (p. 453)**

The thing name.

where used: path; required: true

type: string

CLI

```bash
aws greengrass update-connectivity-info
   --thing-name <value> \
   [--connectivity-info <value>] \
   [--cli-input-json <value>] \
   [--generate-cli-skeleton]
```

cli-input-json format:

```json
{
   "ThingName": "string",
   "ConnectivityInfo": [
```

421
### /greengrass/things/ThingName/connectivityInfo

```json
{
    "Id": "string",
    "HostAddress": "string",
    "PortNumber": "int32-integer",
    "Metadata": "string"
}
```

### Responses

**200**

**success**

**UpdateConnectivityInfoResponse (p. 520)**

```json
{
    "message": "string",
    "Version": "string"
}
```

**message**

A message about the connectivity info update request.

type: string

**Version**

The new version of the connectivity info.

type: string

**400**

invalid request

**GeneralError (p. 476)**

```json
{
    "Message": "string",
    "ErrorDetails": [
    {
        "DetailedErrorCode": "string",
        "DetailedErrorMessage": "string"
    }
    ]
}
```

**GeneralError**

General error information.

type: object

required: ["Message"]

**Message**

A message containing information about the error.

type: string

422
ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

   type: object

DetailedErrorCode

A detailed error code.

   type: string

DetailedErrorMessage

A detailed error message.

   type: string

500

server error

GeneralError (p. 476)

{
   "Message": "string",
   "ErrorDetails": [
      {
         "DetailedErrorCode": "string",
         "DetailedErrorMessage": "string"
      }
   ]
}

GeneralError

General error information.

   type: object

   required: ["Message"]

Message

A message containing information about the error.

   type: string

ErrorDetails

A list of error details.

   type: array

   items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.
type: object

DetailedErrorCode
A detailed error code.
type: string

DetailedErrorMessage
A detailed error message.
type: string

/regrass/updates

POST

POST /regrass/updates

Operation ID: CreateSoftwareUpdateJob (p. 64)

Creates a software update for a core or group of cores (specified as an IoT thing group.) Use this to update the OTA Agent as well as the Greengrass core software. It makes use of the IoT Jobs feature which provides additional commands to manage a Greengrass core software update job.

Produces: application/json

Body Parameters

CreateSoftwareUpdateJobRequestBody (p. 445)

where used: body; required: true

{
  "UpdateTargetsArchitecture": "armv7l|x86_64|aarch64",
  "UpdateTargets": [  
    "string"
  ],
  "SoftwareToUpdate": "core|ota_agent",
  "S3UrlSignerRole": "string",
  "UpdateAgentLogLevel": "NONE|TRACE|DEBUG|VERBOSE|INFO|WARN|ERROR|FATAL",
  "UpdateTargetsOperatingSystem": "ubuntu|rpi|amazon_linux"
}

CreateSoftwareUpdateJobRequest

Request for the CreateSoftwareUpdateJob API.
type: object


UpdateTargetsArchitecture

The architecture of the cores which are the targets of an update.
type: string

enum: ["armv7l", "x86_64", "aarch64"]
UpdateTargets

The ARNs of the targets (IoT things or IoT thing groups) that this update will be applied to.

type: array

SoftwareToUpdate

The piece of software on the Greengrass core that will be updated.

type: string

enum: ["core", "ota_agent"]

S3UrlSignerRole

The IAM Role that Greengrass will use to create pre-signed URLs pointing towards the update artifact.

type: string

UpdateAgentLogLevel

The minimum level of log statements that should be logged by the OTA Agent during an update.

type: string

enum: ["NONE", "TRACE", "DEBUG", "VERBOSE", "INFO", "WARN", "ERROR", "FATAL"]

UpdateTargetsOperatingSystem

The operating system of the cores which are the targets of an update.

type: string

enum: ["ubuntu", "raspbian", "amazon_linux"]

Header Parameters

X-Amzn-Client-Token (p. 455)

A client token used to correlate requests and responses.

where used: header; required: false

type: string

CLI

```
aws greengrass create-software-update-job 
    [--update-targets-architecture <value>] \ 
    [--update-targets <value>] \ 
    [--software-to-update <value>] \ 
    [--s3-url-signer-role <value>] \ 
    [--update-agent-log-level <value>] \ 
    [--update-targets-operating-system <value>] \ 
    [--amzn-client-token <value>] \ 
    [--cli-input-json <value>] \ 
    [--generate-cli-skeleton]
```

cli-input-json format:


```json
{
   "UpdateTargetsArchitecture": "armv7l|x86_64|aarch64",
   "UpdateTargets": [
      "string"
   ],
   "SoftwareToUpdate": "core|ota_agent",
   "S3UrlSignerRole": "string",
   "UpdateAgentLogLevel": "NONE|TRACE|DEBUG|VERBOSE|INFO|WARN|ERROR|FATAL",
   "UpdateTargetsOperatingSystem": "ubuntu|raspbian|amazon_linux",
   "AmznClientToken": "string"
}
```

**Responses**

**200**

success

*CreateSoftwareUpdateJobResponse (p. 462)*

```json
{
   "IotJobId": "string",
   "IotJobArn": "string"
}
```

**CreateSoftwareUpdateJobResponse**

- **type: object**
  
  **IotJobId**
  
  The IoT Job Id corresponding to this update.
  
  - **type: string**
  
  **IotJobArn**
  
  The IoT Job ARN corresponding to this update.
  
  - **type: string**

**400**

invalid request

*GeneralError (p. 476)*

```json
{
   "Message": "string",
   "ErrorDetails": [
      {
         "DetailedErrorCode": "string",
         "DetailedErrorMessage": "string"
      }
   ]
}
```

**GeneralError**

General error information.

- **type: object**
required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

500

server error

GeneralError (p. 476)

{
  "Message": "string",
  "ErrorDetails": [
    {
      "DetailedErrorCode": "string",
      "DetailedErrorMessage": "string"
    }
  ]
}

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.

type: string

ErrorDetails

A list of error details.
Parameters

The following parameters are used:

- `AssociateRoleToGroupRequestBody` (p. 429)
- `AssociateServiceRoleToAccountRequestBody` (p. 429)
- `CertificateAuthorityId` (p. 430)
- `CoreDefinitionId` (p. 430)
- `CoreDefinitionVersionId` (p. 430)
- `CreateCoreDefinitionRequestBody` (p. 431)
- `CreateCoreDefinitionVersionRequestBody` (p. 431)
- `CreateDeploymentRequestBody` (p. 432)
- `CreateDeviceDefinitionRequestBody` (p. 433)
- `CreateDeviceDefinitionVersionRequestBody` (p. 433)
- `CreateFunctionDefinitionRequestBody` (p. 434)
- `CreateFunctionDefinitionVersionRequestBody` (p. 435)
- `CreateGroupRequestBody` (p. 438)
- `CreateGroupVersionRequestBody` (p. 438)
- `CreateLoggerDefinitionRequestBody` (p. 439)
- `CreateLoggerDefinitionVersionRequestBody` (p. 440)
- `CreateResourceDefinitionRequestBody` (p. 441)
- `CreateResourceDefinitionVersionRequestBody` (p. 442)
- `CreateSoftwareUpdateJobRequestBody` (p. 445)
- `CreateSubscriptionDefinitionRequestBody` (p. 446)
- `CreateSubscriptionDefinitionVersionRequestBody` (p. 447)
- `DeploymentId` (p. 448)
- `DeviceDefinitionId` (p. 448)
- `DeviceDefinitionVersionId` (p. 448)
- `FunctionDefinitionId` (p. 449)
- `FunctionDefinitionVersionId` (p. 449)
- `GroupId` (p. 449)
• GroupVersionId (p. 450)
• LoggerDefinitionId (p. 450)
• LoggerDefinitionVersionId (p. 450)
• MaxResults (p. 450)
• NextToken (p. 451)
• ResetDeploymentsRequestBody (p. 451)
• ResourceDefinitionId (p. 452)
• ResourceDefinitionVersionId (p. 452)
• SubscriptionDefinitionId (p. 452)
• SubscriptionDefinitionVersionId (p. 452)
• ThingName (p. 453)
• UpdateConnectivityInfoRequestBody (p. 453)
• UpdateDefinitionRequestBody (p. 454)
• UpdateGroupCertificateConfigurationRequestBody (p. 454)
• X-Amzn-Client-Token (p. 455)

**AssociateRoleToGroupRequestBody**

```json
{
  "RoleArn": "string"
}
```

AssociateRoleToGroupRequestBody

- **in:** body
- **required:** true
- **schema:** AssociateRoleToGroupRequest (p. 457)

**AssociateRoleToGroupRequest**

- **type:** object
- **required:** ["RoleArn"]

**RoleArn**

The ARN of the role you wish to associate with this group.

- **type:** string

**AssociateServiceRoleToAccountRequestBody**

```json
{
  "RoleArn": "string"
}
```

AssociateServiceRoleToAccountRequestBody

- **in:** body
- **required:** true
schema: AssociateServiceRoleToAccountRequest (p. 457)
AssociateServiceRoleToAccountRequest
  type: object
  required: ["RoleArn"]
RoleArn
  The ARN of the service role you wish to associate with your account.
  type: string

CertificateAuthorityId

{
  "CertificateAuthorityId": "string"
}

CertificateAuthorityId
  The ID of the certificate authority.
  in: path
  required: true
  type: string

CoreDefinitionId

{
  "CoreDefinitionId": "string"
}

CoreDefinitionId
  The ID of the core definition.
  in: path
  required: true
  type: string

CoreDefinitionVersionId

{
  "CoreDefinitionVersionId": "string"
}

CoreDefinitionVersionId
  The ID of the core definition version.
CreateCoreDefinitionRequestBody

Information needed to create a core definition.

in: body
required: true

properties

Name: "description"=>"The name of the core definition.", "type"=>"string"

InitialVersion: "$ref"=>"#/definitions/CoreDefinitionVersion", "description"=>"Information about the initial version of the core definition."

CreateCoreDefinitionVersionRequestBody

{ "Cores": [ { "Id": "string", "ThingArn": "string", "CertificateArn": "string", "SyncShadow": true } ] }
CoreDefinitionVersion
  Information about a core definition version.
  type: object

Cores
  A list of cores in the core definition version.
  type: array
  items: Core (p. 458)

Core
  Information about a core.
  type: object
  required: ["Id", "ThingArn", "CertificateArn"]

Id
  The ID of the core.
  type: string

ThingArn
  The ARN of the thing which is the core.
  type: string

CertificateArn
  The ARN of the certificate associated with the core.
  type: string

SyncShadow
  If true, the core's local shadow is automatically synced with the cloud.
  type: boolean

CreateDeploymentRequestBody

```json
{
  "DeploymentType": "NewDeployment|Redeployment|ResetDeployment|ForceResetDeployment",
  "DeploymentId": "string",
  "GroupVersionId": "string"
}
```

CreateDeploymentRequestBody
  in: body
  required: true
  schema: CreateDeploymentRequest (p. 460)

CreateDeploymentRequest
  Information about a deployment.
CreateDeviceDefinitionRequestBody

Information needed to create a device definition.

in: body
required: true
properties

Name: "description"=>"The name of the device definition.", "type"=>"string"

InitialVersion: "$ref"=>"#/definitions/DeviceDefinitionVersion", 
"description"=>"Information about the initial version of the device definition."
CreateFunctionDefinitionRequestBody

```
{
  "ThingArn": "string",
  "CertificateArn": "string",
  "SyncShadow": true
}
```

CreateDeviceDefinitionVersionRequestBody

```
in: body
required: true

schema: DeviceDefinitionVersion (p. 466)
```

DeviceDefinitionVersion

Information about a device definition version.

```
type: object

Devices

A list of devices in the definition version.

```
type: array

items: Device (p. 465)
```

Device

Information about a device.

```
type: object

required: ["Id", "ThingArn", "CertificateArn"]
```

Id

The ID of the device.

```
type: string
```

ThingArn

The thing ARN of the device.

```
type: string
```

CertificateArn

The ARN of the certificate associated with the device.

```
type: string
```

SyncShadow

If true, the device's local shadow will be automatically synced with the cloud.

```
type: boolean
```

CreateFunctionDefinitionRequestBody

```
{
```

434
CreateFunctionDefinitionVersionRequestBody

in: body
required: true

properties

Name: 

"description"=>"The name of the function definition.",
"type"=>"string"

InitialVersion: 

"$ref"=>"#/definitions/FunctionDefinitionVersion",
"description"=>"Information about the initial version of the function definition."

CreateFunctionDefinitionVersionRequest

```json
{
  "Functions": [
    {
      "Id": "string",
      "FunctionArn": "string",
      "FunctionConfiguration": {
        "Pinned": true,
        "Executable": "string",
        "ExecArgs": "string",
        "MemorySize": 0,
        "Timeout": 0,
        "EncodingType": "binary|json",
        "Environment": {
          "Variables": {
            "additionalProperty0": "string",
            "additionalProperty1": "string",
            "additionalProperty2": "string"
          },
          "ResourceAccessPolicies": [
            {
              "ResourceId": "string",
              "Permission": "ro|rw"
            }
          ],
          "AccessSysfs": true
        }
      }
    }
  ]
}
```
CreateFunctionDefinitionVersionRequestBody

Information about the function definition version.

in: body
required: true

schema: FunctionDefinitionVersion (p. 474)

FunctionDefinitionVersion

Information about a function definition version.

type: object

Functions

A list of Lambda functions in this function definition version.

type: array

items: Function (p. 469)

Information about a Lambda function.

type: object

required: ["Id"]

Id

The ID of the Lambda function.

type: string

FunctionArn

The ARN of the Lambda function.

type: string

FunctionConfiguration

The configuration of the Lambda function.

type: object

"Environment": {
  "Variables": {
    "additionalProperty0": "string",
    "additionalProperty1": "string",
    "additionalProperty2": "string"
  },
  "ResourceAccessPolicies": [
    { "ResourceId": "string", "Permission": "ro|rw" }
  ],
  "AccessSysfs": true
}"
Pinned

True if the function is pinned. Pinned means the function is long-lived and starts when the core starts.

type: boolean

Executable

The name of the function executable.

type: string

ExecArgs

The execution arguments.

type: string

MemorySize

The memory size, in KB, which the function requires.

type: integer

Timeout

The allowed function execution time, after which Lambda should terminate the function. This timeout still applies to pinned lambdas for each request.

type: integer

EncodingType

The expected encoding type of the input payload for the function. The default is json.

type: string

enum: ["binary", "json"]

Environment

The environment configuration of the function.

type: object

Variables

Environment variables for the Lambda function's configuration.

type: object

additionalProperties: "{type":"string"}

ResourceAccessPolicies

A list of the resources, with their permissions, to which the Lambda function will be granted access. A Lambda function can have at most 10 resources.

type: array

items: ResourceAccessPolicy (p. 511)

ResourceAccessPolicy

A policy used by the function to access a resource.
CreateGroupRequestBody

- **ResourceId**
  - The ID of the resource. (This ID is assigned to the resource when you create the resource definition.)
  - Type: string

- **Permission**
  - The type of permission a function has to access a resource.
  - Type: string
  - Enum: ["ro", "rw"]

- **AccessSysfs**
  - If true, the Lambda function is allowed to access the host's /sys folder. Use this when the Lambda function needs to read device information from /sys.
  - Type: boolean

CreateGroupRequestBody

```
{
  "Name": "string",
  "InitialVersion": {
    "CoreDefinitionVersionArn": "string",
    "DeviceDefinitionVersionArn": "string",
    "FunctionDefinitionVersionArn": "string",
    "SubscriptionDefinitionVersionArn": "string",
    "LoggerDefinitionVersionArn": "string",
    "ResourceDefinitionVersionArn": "string"
  }
}
```

CreateGroupRequestBody

- **in**: body
- **required**: true
- **properties**
  - **Name**: 
    - description: "The name of the group.", type: "string"
  - **InitialVersion**: 
    - $ref: "#/definitions/GroupVersion", description: "Information about the initial version of the group."

CreateGroupVersionRequestBody

```
{
  "CoreDefinitionVersionArn": "string",
  "DeviceDefinitionVersionArn": "string",
  "FunctionDefinitionVersionArn": "string",
  "SubscriptionDefinitionVersionArn": "string",
  "LoggerDefinitionVersionArn": "string",
  "ResourceDefinitionVersionArn": "string"
}
```
CreateLoggerDefinitionRequestBody

```json
{
  "Name": "string",
  "InitialVersion": {
    "Loggers": [
      {
        "Id": "string",
        "Type": "FileSystem|AWSCloudWatch",
        "Component": "GreengrassSystem|Lambda",
        "Level": "DEBUG|INFO|WARN|ERROR|FATAL",
        "Space": 0
      }
    ]
  }
}
```

CreateGroupVersionRequestBody

```json
in: body
  required: true
  schema: GroupVersion (p. 496)

GroupVersion
  Information about a group version.
  type: object

CoreDefinitionVersionArn
  The ARN of the core definition version for this group.
  type: string

DeviceDefinitionVersionArn
  The ARN of the device definition version for this group.
  type: string

FunctionDefinitionVersionArn
  The ARN of the function definition version for this group.
  type: string

SubscriptionDefinitionVersionArn
  The ARN of the subscription definition version for this group.
  type: string

LoggerDefinitionVersionArn
  The ARN of the logger definition version for this group.
  type: string

ResourceDefinitionVersionArn
  The resource definition version ARN for this group.
  type: string
```

CreateLoggerDefinitionRequestBody
CreateLoggerDefinitionVersionRequestBody

in: body
required: true
properties

Name: "description"=>"The name of the logger definition.", "type"=>"string"

InitialVersion: "$ref"=>"#/definitions/LoggerDefinitionVersion", "description"=>"Information about the initial version of the logger definition."

CreateLoggerDefinitionVersionRequestBody

{
  "Loggers": [
    {
      "Id": "string",
      "Type": "FileSystem|AWSCloudWatch",
      "Component": "GreengrassSystem|Lambda",
      "Level": "DEBUG|INFO|WARN|ERROR|FATAL",
      "Space": 0
    }
  ]
}

CreateLoggerDefinitionVersionRequestBody

in: body
required: true

schema: LoggerDefinitionVersion (p. 506)

LoggerDefinitionVersion

Information about a logger definition version.

type: object

Loggers

A list of loggers.

type: array

items: Logger (p. 505)

Logger

Information about a logger

type: object

required: ["Id", "Type", "Component", "Level"]
Id

The id of the logger.

type: string

Type

type: string

enum: ["FileSystem", "AWSCloudWatch"]

Component

type: string

enum: ["GreengrassSystem", "Lambda"]

Level

type: string

enum: ["DEBUG", "INFO", "WARN", "ERROR", "FATAL"]

Space

The amount of file space, in KB, to use if the local file system is used for logging purposes.

type: integer

CreateResourceDefinitionRequestBody

```json
{
  "Name": "string",
  "InitialVersion": {
    "Resources": [
      {
        "Id": "string",
        "Name": "string",
        "ResourceDataContainer": {
          "*-- NOTE:": "choose one of the following --*",
          "LocalDeviceResourceData": {
            "SourcePath": "string",
            "GroupOwnerSetting": {
              "AutoAddGroupOwner": true,
              "GroupOwner": "string"
            }
          },
          "LocalVolumeResourceData": {
            "SourcePath": "string",
            "DestinationPath": "string",
            "GroupOwnerSetting": {
              "AutoAddGroupOwner": true,
              "GroupOwner": "string"
            }
          },
          "SageMakerMachineLearningModelResourceData": {
            "SageMakerJobArn": "string",
            "DestinationPath": "string"
          },
          "S3MachineLearningModelResourceData": {
            "S3Uri": "string",
            "DestinationPath": "string"
          }
        }
      }
    ]
  }
}
```
CreateResourceDefinitionRequestBody

in: body
required: true
properties

Name: "description"=>"The name of the resource definition.",
"type"=>"string"

InitialVersion: "$ref"=>"#/definitions/ResourceDefinitionVersion",
"description"=>"Information about the initial version of the resource
definition."

CreateResourceDefinitionVersionRequestBody

Information about the resource definition version.
in: body
required: true

schema: ResourceDefinitionVersion (p. 513)

ResourceDefinitionVersion

Information about a resource definition version.

  type: object

Resources

A list of resources.

  type: array

  items: Resource (p. 508)

Resource

Information about a resource.

  type: object

  required: ["Id", "Name", "ResourceDataContainer"]

Id

The resource ID, used to refer to a resource in the Lambda function configuration. Max length is 128 characters with pattern \[a-zA-Z0-9_:]*\+. This must be unique within a Greengrass group.

  type: string

Name

The descriptive resource name, which is displayed on the AWS Greengrass console. Max length 128 characters with pattern \[a-zA-Z0-9_:]*\+. This must be unique within a Greengrass group.

  type: string

ResourceDataContainer

A container for resource data. The container takes only one of the following supported resource data types: LocalDeviceResourceData, LocalVolumeResourceData, SageMakerMachineLearningModelResourceData, S3MachineLearningModelResourceData.

  type: object

  NOTE: oneOf

LocalDeviceResourceData

Attributes that define a local device resource.

  type: object

SourcePath

The local absolute path of the device resource. The source path for a device resource can refer only to a character device or block device under \(/dev\).

  type: string
GroupOwnerSetting

Group owner related settings for local resources.

type: object

AutoAddGroupOwner

If true, AWS Greengrass automatically adds the specified Linux OS group owner of the resource to the Lambda process privileges. Thus the Lambda process will have the file access permissions of the added Linux group.

type: boolean

GroupOwner

The name of the Linux OS group whose privileges will be added to the Lambda process. This field is optional.

type: string

LocalVolumeResourceData

Attributes that define a local volume resource.

type: object

SourcePath

The local absolute path of the volume resource on the host. The source path for a volume resource type cannot start with `/proc` or `/sys`.


type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.


type: string

GroupOwnerSetting

Group owner related settings for local resources.


type: object

AutoAddGroupOwner

If true, AWS Greengrass automatically adds the specified Linux OS group owner of the resource to the Lambda process privileges. Thus the Lambda process will have the file access permissions of the added Linux group.


type: boolean

GroupOwner

The name of the Linux OS group whose privileges will be added to the Lambda process. This field is optional.


type: string

SageMakerMachineLearningModelResourceData

Attributes that define an Amazon SageMaker machine learning resource.


type: object
SageMakerJobArn

The ARN of the Amazon SageMaker training job that represents the source model.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.

type: string

S3MachineLearningModelResourceData

Attributes that define an Amazon S3 machine learning resource.

type: object

S3Uri

The URI of the source model in an Amazon S3 bucket. The model package must be in tar.gz or .zip format.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.

type: string

CreateSoftwareUpdateJobRequestBody

```json
{
  "UpdateTargetsArchitecture": "armv7l|x86_64|aarch64",
  "UpdateTargets": [
    "string"
  ],
  "SoftwareToUpdate": "core|ota_agent",
  "S3UrlSignerRole": "string",
  "UpdateAgentLogLevel": "NONE|TRACE|DEBUG|VERBOSE|INFO|WARN|ERROR|FATAL",
  "UpdateTargetsOperatingSystem": "ubuntu|raspbian|amazon_linux"
}
```

CreateSoftwareUpdateJobRequestBody

in: body
required: true

schema: CreateSoftwareUpdateJobRequest (p. 461)

CreateSoftwareUpdateJobRequest

Request for the CreateSoftwareUpdateJob API.

type: object


UpdateTargetsArchitecture

The architecture of the cores which are the targets of an update.
CreateSubscriptionDefinitionRequestBody

- **Name**: string
- **InitialVersion**: {
  "Subscriptions": [
    {
      "Id": "string",
      "Source": "string",
      "Subject": "string",
      "Target": "string"
    }
  ]
}

CreateSubscriptionDefinitionRequestBody

- in: body
- required: true
CreateSubscriptionDefinitionVersionRequestBody

```json
{
  "Subscriptions": [
    {
      "Id": "string",
      "Source": "string",
      "Subject": "string",
      "Target": "string"
    }
  ]
}
```

CreateSubscriptionDefinitionVersionRequestBody

- `in: body`
- `required: true`
- `schema: SubscriptionDefinitionVersion (p. 518)`

SubscriptionDefinitionVersion

- Information about a subscription definition version.
  - `type: object`

Subscriptions

- A list of subscriptions.
  - `type: array`

Subscription

- Information about a subscription.
  - `type: object`

  - `required: ['Id', 'Source', 'Subject', 'Target']`

  Id

  - The id of the subscription.
    - `type: string`

  Source

  - The source of the subscription. Can be a thing ARN, a Lambda function ARN, 'cloud' (which represents the AWS IoT cloud), or 'GGShadowService'.
type: string

Subject

The subject of the message.

type: string

Target

Where the message is sent to. Can be a thing ARN, a Lambda function ARN, 'cloud' (which represents the AWS IoT cloud), or 'GGShadowService'.

type: string

DeploymentId

```
{
  "DeploymentId": "string"
}
```

DeploymentId

The ID of the deployment.

- in: path
- required: true
- type: string

DeviceDefinitionId

```
{
  "DeviceDefinitionId": "string"
}
```

DeviceDefinitionId

The ID of the device definition.

- in: path
- required: true
- type: string

DeviceDefinitionVersionId

```
{
  "DeviceDefinitionVersionId": "string"
}
```

DeviceDefinitionVersionId

The ID of the device definition version.
FunctionDefinitionId

{
  "FunctionDefinitionId": "string"
}

FunctionDefinitionId

The ID of the Lambda function definition.

in: path
required: true
type: string

FunctionDefinitionVersionId

{
  "FunctionDefinitionVersionId": "string"
}

FunctionDefinitionVersionId

The ID of the function definition version.

in: path
required: true
type: string

GroupId

{
  "GroupId": "string"
}

GroupId

The ID of the AWS Greengrass group.

in: path
required: true
type: string
**GroupVersionId**

```json
{
    "GroupVersionId": "string"
}
```

*GroupVersionId*

The ID of the group version.

- in: path
- required: true
- type: string

**LoggerDefinitionId**

```json
{
    "LoggerDefinitionId": "string"
}
```

*LoggerDefinitionId*

The ID of the logger definition.

- in: path
- required: true
- type: string

**LoggerDefinitionVersionId**

```json
{
    "LoggerDefinitionVersionId": "string"
}
```

*LoggerDefinitionVersionId*

The ID of the logger definition version.

- in: path
- required: true
- type: string

**MaxResults**

```json
{
    "MaxResults": 100
}
```
MaxResults

The maximum number of results to be returned per request.
- **in:** query
- **required:** false
- **min:** 1
- **max:** 250
- **default:** 250
- **type:** integer

### NextToken

| { |
| "NextToken": "string" |
|

- **NextToken**
  - The token for the next set of results, or null if there are no additional results.
  - **in:** query
  - **required:** false
  - **type:** string

### ResetDeploymentsRequestBody

| { |
| "Force": true |
|

- **ResetDeploymentsRequestBody**
  - Information needed to reset deployments.
  - **in:** body
  - **required:** true
  - **schema:** [ResetDeploymentsRequest](p. 507)

### ResetDeploymentsRequest

- **ResetDeploymentsRequest**
  - Information about a group reset request.
  - **type:** object

- **Force**
  - If true, performs a best-effort only core reset.
  - **type:** boolean
ResourceDefinitionId

```json
{
  "ResourceDefinitionId": "string"
}
```

**ResourceDefinitionId**

The ID of the resource definition.

- **in:** path
- **required:** true
- **type:** string

ResourceDefinitionVersionId

```json
{
  "ResourceDefinitionVersionId": "string"
}
```

**ResourceDefinitionVersionId**

The ID of the resource definition version.

- **in:** path
- **required:** true
- **type:** string

SubscriptionDefinitionId

```json
{
  "SubscriptionDefinitionId": "string"
}
```

**SubscriptionDefinitionId**

The ID of the subscription definition.

- **in:** path
- **required:** true
- **type:** string

SubscriptionDefinitionVersionId

```json
{
  "SubscriptionDefinitionVersionId": "string"
}
```

**SubscriptionDefinitionVersionId**
SubscriptionDefinitionVersionId

The ID of the subscription definition version.

in: path
required: true
type: string

ThingName

The thing name.
in: path
required: true
type: string

UpdateConnectivityInfoRequestBody

The connectivity information.
in: body

schema: UpdateConnectivityInfoRequest (p. 520)

Information required to update a Greengrass core's connectivity.

ConnectivityInfo

A list of connectivity info.
type: array
items: ConnectivityInfo (p. 458)

Information about a Greengrass core's connectivity.

required: ["Id", "HostAddress"]

Id

The ID of the connectivity information.

type: string

HostAddress

The endpoint for the Greengrass core. Can be an IP address or DNS.

type: string

PortNumber

The port of the Greengrass core. Usually 8883.

type: integer

format: int32

Metadata

Metadata for this endpoint.

type: string

UpdateDefinitionRequestBody

{
  "Name": "string"
}

UpdateDefinitionRequestBody

in: body

required: true

properties

Name: "description"=>"The name of the definition.", "required"=>true,
  "type"=>"string"

UpdateGroupCertificateConfigurationRequestBody

{
  "CertificateExpiryInMilliseconds": "string"
}

UpdateGroupCertificateConfigurationRequestBody

in: body

required: true
schema: [UpdateGroupCertificateConfigurationRequest](#) (p. 523)

updateGroupCertificateConfigurationRequest

  type: object

  required: ["CertificateExpiryInMilliseconds"]

CertificateExpiryInMilliseconds

  The amount of time remaining before the certificate expires, in milliseconds.

  type: string

X-Amzn-Client-Token

```json
{
  "X-Amzn-Client-Token": "string"
}
```

X-Amzn-Client-Token

  A client token used to correlate requests and responses.

  in: header

  type: string

  required: false

Definitions

The following definitions are used:

- [AssociateRoleToGroupRequest](#) (p. 457)
- [AssociateServiceRoleToAccountRequest](#) (p. 457)
- [AssociateServiceRoleToAccountResponse](#) (p. 458)
- [ConnectivityInfo](#) (p. 458)
- [Core](#) (p. 458)
- [CoreDefinitionVersion](#) (p. 459)
- [CreateDeploymentRequest](#) (p. 460)
- [CreateDeploymentResponse](#) (p. 461)
- [CreateGroupCertificateAuthorityResponse](#) (p. 461)
- [CreateSoftwareUpdateJobRequest](#) (p. 461)
- [CreateSoftwareUpdateJobResponse](#) (p. 462)
- [DefinitionInformation](#) (p. 463)
- [Deployment](#) (p. 464)
- [DeploymentType](#) (p. 464)
- [Deployments](#) (p. 464)
- [Device](#) (p. 465)
- [DeviceDefinitionVersion](#) (p. 466)
- [DisassociateRoleFromGroupResponse](#) (p. 467)
- [DisassociateServiceRoleFromAccountResponse](#) (p. 467)
• Empty Schema (p. 467)
• ErrorDetail (p. 468)
• ErrorDetails (p. 468)
• Function (p. 469)
• FunctionConfiguration (p. 471)
• FunctionConfigurationEnvironment (p. 473)
• FunctionDefinitionVersion (p. 474)
• GeneralError (p. 476)
• GetAssociatedRoleResponse (p. 477)
• GetConnectivityInfoResponse (p. 477)
• GetCoreDefinitionVersionResponse (p. 478)
• GetDeploymentStatusResponse (p. 480)
• GetDeviceDefinitionVersionResponse (p. 481)
• GetFunctionDefinitionVersionResponse (p. 482)
• GetGroupCertificateAuthorityResponse (p. 485)
• GetGroupVersionResponse (p. 486)
• GetLoggerDefinitionVersionResponse (p. 487)
• GetResourceDefinitionVersionResponse (p. 489)
• GetServiceRoleForAccountResponse (p. 492)
• GetSubscriptionDefinitionVersionResponse (p. 492)
• Group (p. 494)
• Group Certificate Authority Properties (p. 494)
• GroupCertificateConfiguration (p. 495)
• GroupInformation (p. 495)
• GroupOwnerSetting (p. 496)
• GroupVersion (p. 496)
• IDList (p. 497)
• ListDefinitionsResponse (p. 498)
• ListDeploymentsRequest (p. 499)
• ListDeploymentsResponse (p. 499)
• ListGroupCertificateAuthoritiesResponse (p. 500)
• ListGroupsResponse (p. 501)
• ListVersionsResponse (p. 502)
• LocalDeviceResourceData (p. 503)
• LocalVolumeResourceData (p. 504)
• Logger (p. 505)
• LoggerComponent (p. 506)
• LoggerDefinitionVersion (p. 506)
• LoggerLevel (p. 507)
• LoggerType (p. 507)
• Permission (p. 507)
• ResetDeploymentsRequest (p. 507)
• ResetDeploymentsResponse (p. 508)
• Resource (p. 508)
• ResourceAccessPolicy (p. 511)
• ResourceDataContainer (p. 511)
AssociateRoleToGroupRequest

```json
{
   "RoleArn": "string"
}
```

**AssociateRoleToGroupRequest**

- **type**: object
- **required**: `"RoleArn"`

**RoleArn**

The ARN of the role you wish to associate with this group.

- **type**: string

AssociateServiceRoleToAccountRequest

```json
{
   "RoleArn": "string"
}
```

**AssociateServiceRoleToAccountRequest**

- **type**: object
- **required**: `"RoleArn"`

**RoleArn**

The ARN of the service role you wish to associate with your account.

- **type**: string
AssociateServiceRoleToAccountResponse

```json
{
   "AssociatedAt": "string"
}
```

**AssociateServiceRoleToAccountResponse**

- **type**: object  
- **AssociatedAt**
  - The time when the service role was associated with the account.
  - **type**: string

**ConnectivityInfo**

```json
{
   "Id": "string",
   "HostAddress": "string",
   "PortNumber": 0x01,
   "Metadata": "string"
}
```

- Information about a Greengrass core's connectivity.
- **required**: ["Id", "HostAddress"]
- **Id**
  - The ID of the connectivity information.
  - **type**: string
- **HostAddress**
  - The endpoint for the Greengrass core. Can be an IP address or DNS.
  - **type**: string
- **PortNumber**
  - The port of the Greengrass core. Usually 8883.
  - **type**: integer
  - **format**: int32
- **Metadata**
  - Metadata for this endpoint.
  - **type**: string

**Core**

```json
{
}
```
Core

Information about a core.

type: object

required: ["Id", "ThingArn", "CertificateArn"]

Id

The ID of the core.

type: string

ThingArn

The ARN of the thing which is the core.

type: string

CertificateArn

The ARN of the certificate associated with the core.

type: string

SyncShadow

If true, the core's local shadow is automatically synced with the cloud.

type: boolean

CoreDefinitionVersion

{
  "Cores": [
    {
      "Id": "string",
      "ThingArn": "string",
      "CertificateArn": "string",
      "SyncShadow": true
    }
  ]
}

CoreDefinitionVersion

Information about a core definition version.

type: object

Cores

A list of cores in the core definition version.

type: array

items: Core (p. 458)
Core

Information about a core.

type: object

required: ["Id", "ThingArn", "CertificateArn"]

Id

The ID of the core.

type: string

ThingArn

The ARN of the thing which is the core.

type: string

CertificateArn

The ARN of the certificate associated with the core.

type: string

SyncShadow

If true, the core's local shadow is automatically synced with the cloud.

type: boolean

CreateDeploymentRequest

{
  "DeploymentType": "NewDeployment|Redeployment|ResetDeployment|ForceResetDeployment",
  "DeploymentId": "string",
  "GroupVersionId": "string"
}

CreateDeploymentRequest

Information about a deployment.

type: object

required: ["DeploymentType"]

DeploymentType

type: string

enum: ["NewDeployment", "Redeployment", "ResetDeployment", "ForceResetDeployment"]

DeploymentId

The ID of the deployment if you wish to redeploy a previous deployment.

type: string

GroupVersionId

The ID of the group version to be deployed.

type: string
CreateDeploymentResponse

```json
{
  "DeploymentArn": "string",
  "DeploymentId": "string"
}
```

CreateDeploymentResponse
- type: object
- DeploymentArn
  - The ARN of the deployment.
  - type: string
- DeploymentId
  - The ID of the deployment.
  - type: string

CreateGroupCertificateAuthorityResponse

```json
{
  "GroupCertificateAuthorityArn": "string"
}
```

CreateGroupCertificateAuthorityResponse
- type: object
  - GroupCertificateAuthorityArn
    - The ARN of the group certificate authority.
    - type: string

CreateSoftwareUpdateJobRequest

```json
{
  "UpdateTargetsArchitecture": "armv7l|x86_64|aarch64",
  "UpdateTargets": [
    "string"
  ],
  "SoftwareToUpdate": "core|ota_agent",
  "S3UrlSignerRole": "string",
  "UpdateAgentLogLevel": "NONE|TRACE|DEBUG|VERBOSE|INFO|WARN|ERROR|FATAL",
  "UpdateTargetsOperatingSystem": "ubuntu|raspbian|amazon_linux"
}
```

CreateSoftwareUpdateJobRequest
- Request for the CreateSoftwareUpdateJob API.
- type: object

UpdateTargetsArchitecture
The architecture of the cores which are the targets of an update.

type: string
enum: ["armv7l", "x86_64", "aarch64"]

UpdateTargets
The ARNs of the targets (IoT things or IoT thing groups) that this update will be applied to.

type: array

SoftwareToUpdate
The piece of software on the Greengrass core that will be updated.

type: string
enum: ["core", "ota_agent"]

S3UrlSignerRole
The IAM Role that Greengrass will use to create pre-signed URLs pointing towards the update artifact.

type: string

UpdateAgentLogLevel
The minimum level of log statements that should be logged by the OTA Agent during an update.

type: string
enum: ["NONE", "TRACE", "DEBUG", "VERBOSE", "INFO", "WARN", "ERROR", "FATAL"]

UpdateTargetsOperatingSystem
The operating system of the cores which are the targets of an update.

type: string
enum: ["ubuntu", "raspbian", "amazon_linux"]

---

CreateSoftwareUpdateJobResponse

```json
{
    "IotJobId": "string",
    "IotJobArn": "string"
}
```

CreateSoftwareUpdateJobResponse
type: object

IotJobId
The IoT Job Id corresponding to this update.

type: string
IoTJobArn

The IoT Job ARN corresponding to this update.

type: string

DefinitionInformation

{
"Name": "string",
"Id": "string",
"Arn": "string",
"LastUpdatedTimestamp": "string",
"CreationTimestamp": "string",
"LatestVersion": "string",
"LatestVersionArn": "string"
}

DefinitionInformation

Information about a definition.

type: object

Name

The name of the definition.

type: string

Id

The ID of the definition.

type: string

Arn

The ARN of the definition.

type: string

LastUpdatedTimestamp

The time, in milliseconds since the epoch, when the definition was last updated.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the definition was created.

type: string

LatestVersion

The latest version of the definition.

type: string

LatestVersionArn

The ARN of the latest version of the definition.

type: string
Deployment

```json
{
    "GroupArn": "string",
    "DeploymentId": "string",
    "DeploymentArn": "string",
    "DeploymentType": "NewDeployment|Redeployment|ResetDeployment|ForceResetDeployment",
    "CreatedAt": "string"
}
```

Deployment

Information about a deployment.

**type:** object

**GroupArn**

The ARN of the group for this deployment.

**type:** string

**DeploymentId**

The ID of the deployment.

**type:** string

**DeploymentArn**

The ARN of the deployment.

**type:** string

**DeploymentType**

**type:** string

**enum:** ["NewDeployment", "Redeployment", "ResetDeployment", "ForceResetDeployment"]

**CreatedAt**

The time, in milliseconds since the epoch, when the deployment was created.

**type:** string

DeploymentType

```json
{}
```

**type:** string

**enum:** ["NewDeployment", "Redeployment", "ResetDeployment", "ForceResetDeployment"]

Depoyments

```json
[
]
```
Device

```json
{
    "Id": "string",
    "ThingArn": "string",
    "CertificateArn": "string",
    "SyncShadow": true
}
```

Device

Information about a device.

type: object
DeviceDefinitionVersion

```
{
  "Devices": [
    {
      "Id": "string",
      "ThingArn": "string",
      "CertificateArn": "string",
      "SyncShadow": true
    }
  ]
}
```

DeviceDefinitionVersion

Information about a device definition version.

type: object

Devices

A list of devices in the definition version.

type: array

items: Device (p. 465)

Device

Information about a device.

type: object

required: ["Id", "Arn", "CertificateArn"]

Id

The ID of the device.
DisassociateRoleFromGroupResponse

- **type**: string
- **ThingArn**
  - The thing ARN of the device.
  - **type**: string
- **CertificateArn**
  - The ARN of the certificate associated with the device.
  - **type**: string
- **SyncShadow**
  - If true, the device's local shadow will be automatically synced with the cloud.
  - **type**: boolean

```json
{
   "DisassociatedAt": "string"
}
```

DisassociateServiceRoleFromAccountResponse

- **type**: object
- **DisassociatedAt**
  - The time, in milliseconds since the epoch, when the role was disassociated from the group.
  - **type**: string

```json
{
   "DisassociatedAt": "string"
}
```

Empty Schema

```json
{
}
```
Empty Schema

Empty

type: object

ErrorDetail

{  "DetailedErrorCode": "string",  "DetailedErrorMessage": "string"}

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

ErrorDetails

[  {  "DetailedErrorCode": "string",  "DetailedErrorMessage": "string"  }
]

ErrorDetails

A list of error details.

type: array

items: ErrorDetail (p. 468)

ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string
DetailedErrorMessage

A detailed error message.

type: string

Function

```json
{
  "Id": "string",
  "FunctionArn": "string",
  "FunctionConfiguration": {
    "Pinned": true,
    "Executable": "string",
    "ExecArgs": "string",
    "MemorySize": 0,
    "Timeout": 0,
    "EncodingType": "binary|json",
    "Environment": {
      "Variables": {
        "additionalProperty0": "string",
        "additionalProperty1": "string",
        "additionalProperty2": "string"
      },
      "ResourceAccessPolicies": [
        {
          "ResourceId": "string",
          "Permission": "ro|rw"
        }
      ],
      "AccessSysfs": true
    }
  }
}
```

Information about a Lambda function.


type: object

required: ["Id"]

Id

The ID of the Lambda function.


type: string

FunctionArn

The ARN of the Lambda function.


type: string

FunctionConfiguration

The configuration of the Lambda function.


type: object

Pinned

True if the function is pinned. Pinned means the function is long-lived and starts when the core starts.
**Function**

- **Executable**
  - The name of the function executable.
  - **type**: string

- **ExecArgs**
  - The execution arguments.
  - **type**: string

- **MemorySize**
  - The memory size, in KB, which the function requires.
  - **type**: integer

- **Timeout**
  - The allowed function execution time, after which Lambda should terminate the function. This timeout still applies to pinned lambdas for each request.
  - **type**: integer

- **EncodingType**
  - The expected encoding type of the input payload for the function. The default is `json`.
  - **type**: string
  - **enum**: ["binary", "json"]

- **Environment**
  - The environment configuration of the function.
  - **type**: object

- **Variables**
  - Environment variables for the Lambda function's configuration.
  - **type**: object
  - **additionalProperties**: "type"=>"string"

- **ResourceAccessPolicies**
  - A list of the resources, with their permissions, to which the Lambda function will be granted access. A Lambda function can have at most 10 resources.
  - **type**: array
  - **items**: [ResourceAccessPolicy (p. 511)]

- **ResourceAccessPolicy**
  - A policy used by the function to access a resource.
  - **type**: object
  - **required**: ["ResourceId"]

- **ResourceId**
  - The ID of the resource. (This ID is assigned to the resource when you create the resource definiton.)
FunctionConfiguration

```json
{
  "Pinned": true,
  "Executable": "string",
  "ExecArgs": "string",
  "MemorySize": 0,
  "Timeout": 0,
  "EncodingType": "binary|json",
  "Environment": {
    "Variables": {
      "additionalProperty0": "string",
      "additionalProperty1": "string",
      "additionalProperty2": "string"
    },
    "ResourceAccessPolicies": [
      {
        "ResourceId": "string",
        "Permission": "ro|rw"
      }
    ],
    "AccessSysfs": true
  }
}
```

The configuration of the Lambda function.

- **Pinned**: object
  - **Pinned**: boolean
    - True if the function is pinned. Pinned means the function is long-lived and starts when the core starts.
  - **Executable**: string
    - The name of the function executable.
  - **ExecArgs**: string
    - The execution arguments.
type: string
MemorySize

The memory size, in KB, which the function requires.

type: integer
Timeout

The allowed function execution time, after which Lambda should terminate the function. This timeout still applies to pinned lambdas for each request.

type: integer
EncodingType

The expected encoding type of the input payload for the function. The default is json.

type: string
enum: ["binary", "json"]
Environment

The environment configuration of the function.

type: object
Variables

Environment variables for the Lambda function's configuration.

type: object
additionalProperties: "type"=>"string"
ResourceAccessPolicies

A list of the resources, with their permissions, to which the Lambda function will be granted access. A Lambda function can have at most 10 resources.

type: array
items: ResourceAccessPolicy (p. 511)
ResourceAccessPolicy

A policy used by the function to access a resource.

type: object
required: ["ResourceId"]
ResourceId

The ID of the resource. (This ID is assigned to the resource when you create the resource definition.)

type: string
Permission

The type of permission a function has to access a resource.

type: string
enum: ["ro", "rw"]
AccessSysfs

If true, the Lambda function is allowed to access the host's /sys folder. Use this when the Lambda function needs to read device information from /sys.

type: boolean

FunctionConfigurationEnvironment

```
{
    "Variables": {
        "additionalProperty0": "string",
        "additionalProperty1": "string",
        "additionalProperty2": "string"
    },
    "ResourceAccessPolicies": [
        { "ResourceId": "string", "Permission": "ro|rw" }
    ],
    "AccessSysfs": true
}
```

The environment configuration of the function.

type: object

Variables

Environment variables for the Lambda function's configuration.

type: object

additionalProperties: "type"=>"string"

ResourceAccessPolicies

A list of the resources, with their permissions, to which the Lambda function will be granted access. A Lambda function can have at most 10 resources.

type: array

items: ResourceAccessPolicy (p. 511)

ResourceAccessPolicy

A policy used by the function to access a resource.

type: object

required: ["ResourceId"]

ResourceId

The ID of the resource. (This ID is assigned to the resource when you create the resource definition.)

type: string

Permission

The type of permission a function has to access a resource.
FunctionDefinitionVersion

Information about a function definition version.

type: object

Functions

A list of Lambda functions in this function definition version.

type: array

items: Function (p. 469)

Information about a Lambda function.

type: object
required: ["Id"]

Id

The ID of the Lambda function.

type: string

FunctionArn

The ARN of the Lambda function.

type: string

FunctionConfiguration

The configuration of the Lambda function.

type: object

Pinned

True if the function is pinned. Pinned means the function is long-lived and starts when the core
starts.

type: boolean

Executable

The name of the function executable.

type: string

ExecArgs

The execution arguments.

type: string

MemorySize

The memory size, in KB, which the function requires.

type: integer

Timeout

The allowed function execution time, after which Lambda should terminate the function. This
timeout still applies to pinned lambdas for each request.

type: integer

EncodingType

The expected encoding type of the input payload for the function. The default is json.

type: string

enum: ["binary", "json"]

Environment

The environment configuration of the function.

type: object

Variables

Environment variables for the Lambda function's configuration.

type: object
additionalProperties: "type=>"string"

ResourceAccessPolicies

A list of the resources, with their permissions, to which the Lambda function will be granted access. A Lambda function can have at most 10 resources.

type: array

items: ResourceAccessPolicy (p. 511)

ResourceAccessPolicy

A policy used by the function to access a resource.

type: object

required: ["ResourceId"]

ResourceId

The ID of the resource. (This ID is assigned to the resource when you create the resource definition.)

type: string

Permission

The type of permission a function has to access a resource.

type: string

enum: ["ro", "rw"]

AccessSysfs

If true, the Lambda function is allowed to access the host’s /sys folder. Use this when the Lambda function needs to read device information from /sys.

type: boolean

GeneralError

```json
{
"Message": "string",
"ErrorDetails": [
{
"DetailedErrorCode": "string",
"DetailedErrorMessage": "string"
}
]
}
```

GeneralError

General error information.

type: object

required: ["Message"]

Message

A message containing information about the error.
GetAssociatedRoleResponse

```
{  
  "AssociatedAt": "string",
  "RoleArn": "string"
}
```

**GetAssociatedRoleResponse**

type: object

**AssociatedAt**

The time when the role was associated with the group.

type: string

**RoleArn**

The ARN of the role that is associated with the group.

type: string

GetConnectivityInfoResponse

```
{
  "message": "string",
  "ConnectivityInfo": [
  {
    "Id": "string",
    "HostAddress": "string",
    "PortNumber": 0x01,
    "Metadata": "string"
  }
}
```

**GetConnectivityInfoResponse**

type: object

**message**

A list of error details.

type: array

**ErrorDetails**

Details about the error.

type: object

**DetailedErrorCode**

A detailed error code.

type: string

**DetailedErrorMessage**

A detailed error message.

type: string
GetCoreDefinitionVersionResponse

{{
  "Arn": "string",
  "Id": "string",
  "Version": "string",
  "CreationTimestamp": "string",
  "Definition": {
    "Cores": [
      {
        "Id": "string",
        "ThingArn": "string",
        "CertificateArn": "string",
      }
    ]
  }
}}
"SyncShadow": true
}
]
}
}

GetCoreDefinitionVersionResponse

type: object

Arn

The ARN of the core definition version.

type: string

Id

The ID of the core definition version.

type: string

Version

The version of the core definition version.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the core definition version was created.

type: string

Definition

Information about a core definition version.

type: object

Cores

A list of cores in the core definition version.

type: array

items: Core (p. 458)

Core

Information about a core.

type: object

required: ["Id", "ThingArn", "CertificateArn"]

Id

The ID of the core.

type: string

ThingArn

The ARN of the thing which is the core.

type: string

479
CertificateArn
   The ARN of the certificate associated with the core.
   type: string

SyncShadow
   If true, the core's local shadow is automatically synced with the cloud.
   type: boolean

GetDeploymentStatusResponse

{
   "DeploymentStatus": "string",
   "DeploymentType": "NewDeployment|Redeployment|ResetDeployment|ForceResetDeployment",
   "UpdatedAt": "string",
   "ErrorMessage": "string",
   "ErrorDetails": [
   {  
       "DetailedErrorCode": "string",
       "DetailedErrorMessage": "string"
   }
   ]
}

GetDeploymentStatusResponse
   Information about the status of a deployment for a group.
   type: object

DeploymentStatus
   The status of the deployment.
   type: string

DeploymentType
   type: string
   enum: ["NewDeployment", "Redeployment", "ResetDeployment", "ForceResetDeployment"]

UpdatedAt
   The time, in milliseconds since the epoch, when the deployment status was updated.
   type: string

ErrorMessage
   Error message
   type: string

ErrorDetails
   A list of error details.
   type: array
   items: ErrorDetail (p. 468)
ErrorDetail

Details about the error.

type: object

DetailedErrorCode

A detailed error code.

type: string

DetailedErrorMessage

A detailed error message.

type: string

GetDeviceDefinitionVersionResponse

```json
{
    "Arn": "string",
    "Id": "string",
    "Version": "string",
    "CreationTimestamp": "string",
    "Definition": {
        "Devices": [
            {
                "Id": "string",
                "ThingArn": "string",
                "CertificateArn": "string",
                "SyncShadow": true
            }
        ]
    }
}
```

GetDeviceDefinitionVersionResponse

type: object

Arn

The ARN of the device definition version.

   type: string

Id

The ID of the device definition version.

   type: string

Version

The version of the device definition version.

   type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the device definition version was created.

   type: string
Definition
   Information about a device definition version.
   type: object

Devices
   A list of devices in the definition version.
   type: array
   items: Device (p. 465)

Device
   Information about a device.
   type: object
   required: ["Id", "ThingArn", "CertificateArn"]

Id
   The ID of the device.
   type: string

ThingArn
   The thing ARN of the device.
   type: string

CertificateArn
   The ARN of the certificate associated with the device.
   type: string

SyncShadow
   If true, the device's local shadow will be automatically synced with the cloud.
   type: boolean

GetFunctionDefinitionVersionResponse

{
   "Arn": "string",
   "Id": "string",
   "Version": "string",
   "CreationTimestamp": "string",
   "Definition": {
      "Functions": [
         {
            "Id": "string",
            "FunctionArn": "string",
            "FunctionConfiguration": {
               "Pinned": true,
               "Executable": "string",
               "ExecArgs": "string",
               "MemorySize": 0,
               "Timeout": 0,
               "EncodingType": "binary|json",
            }
         }
      ]
   }
}

482
GetFunctionDefinitionVersionResponse

Information about a function definition version.

type: object

Arn

The ARN of the function definition version.

type: string

Id

The ID of the function definition version.

type: string

Version

The version of the function definition version.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the function definition version was created.

type: string

Definition

Information about a function definition version.

type: object

Functions

A list of Lambda functions in this function definition version.

type: array

items: Function (p. 469)

Information about a Lambda function.

type: object
required: ["Id"]

Id
The ID of the Lambda function.

type: string

FunctionArn
The ARN of the Lambda function.

type: string

FunctionConfiguration
The configuration of the Lambda function.

type: object

Pinned
True if the function is pinned. Pinned means the function is long-lived and starts when the core starts.

type: boolean

Executable
The name of the function executable.

type: string

ExecArgs
The execution arguments.

type: string

MemorySize
The memory size, in KB, which the function requires.

type: integer

Timeout
The allowed function execution time, after which Lambda should terminate the function. This timeout still applies to pinned lambdas for each request.

type: integer

EncodingType
The expected encoding type of the input payload for the function. The default is json.

type: string

enum: ["binary", "json"]

Environment
The environment configuration of the function.

type: object

Variables
Environment variables for the Lambda function's configuration.
GetGroupCertificateAuthorityResponse

Information about a certificate authority for a group.

type: object

PemEncodedCertificate

The PEM encoded certificate for the group.

type: string

GroupCertificateAuthorityArn

The ARN of the certificate authority for the group.
GroupCertificateAuthorityId

The ID of the certificate authority for the group.

type: string

GetGroupVersionResponse

Information about a group version.

type: object

Arn

The ARN of the group version.

type: string

Id

The ID of the group version.

type: string

Version

The unique ID for the version of the group.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the group version was created.

type: string

Definition

Information about a group version.

type: object

CoreDefinitionVersionArn

The ARN of the core definition version for this group.
GetLoggerDefinitionVersionResponse

Information about a logger definition version.

- **Arn**: The ARN of the logger definition version.
- **Id**: string
- **Version**: string
- **CreationTimestamp**: string
- **Definition**: object
  - **Loggers**: array
    - **Id**: string
    - **Type**: FileSystem|AWSCloudWatch
    - **Component**: GreengrassSystem|Lambda
    - **Level**: DEBUG|INFO|WARN|ERROR|FATAL
    - **Space**: 0
Id
The ID of the logger definition version.
type: string

Version
The version of the logger definition version.
type: string

CreationTimestamp
The time, in milliseconds since the epoch, when the logger definition version was created.
type: string

Definition
Information about a logger definition version.
type: object

Loggers
A list of loggers.
type: array
items: Logger (p. 505)

Logger
Information about a logger
type: object
required: ["Id", "Type", "Component", "Level"]

Id
The id of the logger.
type: string

Type
type: string
enum: ["FileSystem", "AWSCloudWatch"]

Component
type: string
enum: ["GreengrassSystem", "Lambda"]

Level
type: string
enum: ["DEBUG", "INFO", "WARN", "ERROR", "FATAL"]

Space
The amount of file space, in KB, to use if the local file system is used for logging purposes.
type: integer
GetResourceDefinitionVersionResponse

Information about a resource definition version.

type: object

Arn

Arn of the resource definition version.

type: string

Id

The ID of the resource definition version.

type: string

Version

The version of the resource definition version.
type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the resource definition version was created.

type: string

Definition

Information about a resource definition version.

type: object

Resources

A list of resources.

type: array

items: Resource (p. 508)

Resource

Information about a resource.

type: object

required: ["Id", "Name", "ResourceDataContainer"]

Id

The resource ID, used to refer to a resource in the Lambda function configuration. Max length is 128 characters with pattern \[a-zA-Z0-9:_-]+. This must be unique within a Greengrass group.

type: string

Name

The descriptive resource name, which is displayed on the AWS Greengrass console. Max length 128 characters with pattern \[a-zA-Z0-9:_-]+. This must be unique within a Greengrass group.

type: string

ResourceDataContainer

A container for resource data. The container takes only one of the following supported resource data types: LocalDeviceResourceData, LocalVolumeResourceData, SageMakerMachineLearningModelResourceData, S3MachineLearningModelResourceData.

type: object

NOTE: oneOf

LocalDeviceResourceData

Attributes that define a local device resource.

type: object

SourcePath

The local absolute path of the device resource. The source path for a device resource can refer only to a character device or block device under /dev.

type: string
GroupOwnerSetting

Group owner related settings for local resources.

type: object

AutoAddGroupOwner

If true, AWS Greengrass automatically adds the specified Linux OS group owner of the resource to the Lambda process privileges. Thus the Lambda process will have the file access permissions of the added Linux group.

type: boolean

GroupOwner

The name of the Linux OS group whose privileges will be added to the Lambda process. This field is optional.

type: string

LocalVolumeResourceData

Attributes that define a local volume resource.

type: object

SourcePath

The local absolute path of the volume resource on the host. The source path for a volume resource type cannot start with /proc or /sys.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.

type: string

GroupOwnerSetting

Group owner related settings for local resources.

type: object

AutoAddGroupOwner

If true, AWS Greengrass automatically adds the specified Linux OS group owner of the resource to the Lambda process privileges. Thus the Lambda process will have the file access permissions of the added Linux group.

type: boolean

GroupOwner

The name of the Linux OS group whose privileges will be added to the Lambda process. This field is optional.

type: string

SageMakerMachineLearningModelResourceData

Attributes that define an Amazon SageMaker machine learning resource.

type: object
SageMakerJobArn

The ARN of the Amazon SageMaker training job that represents the source model.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.

type: string

S3MachineLearningModelResourceData

Attributes that define an Amazon S3 machine learning resource.

type: object

S3Uri

The URI of the source model in an Amazon S3 bucket. The model package must be in tar.gz or .zip format.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.

type: string

GetServiceRoleForAccountResponse

{
  "AssociatedAt": "string",
  "RoleArn": "string"
}

GetServiceRoleForAccountResponse

type: object

AssociatedAt

The time when the service role was associated with the account.

type: string

RoleArn

The ARN of the role which is associated with the account.

type: string

GetSubscriptionDefinitionVersionResponse

{
  "Arn": "string",
  "Id": "string",
  "Version": "string",
  "CreationTimestamp": "string",
  "Definition": {

"Arn": "string",
"Id": "string",
"Version": "string",
"CreationTimestamp": "string",
"Definition": {

```
"Subscriptions": [  
  {  
    "Id": "string",  
    "Source": "string",  
    "Subject": "string",  
    "Target": "string"  
  }
]

GetSubscriptionDefinitionVersionResponse

Information about a subscription definition version.

type: object

Arn

The ARN of the subscription definition version.

type: string

Id

The ID of the subscription definition version.

type: string

Version

The version of the subscription definition version.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the subscription definition version was created.

type: string

Definition

Information about a subscription definition version.

type: object

Subscriptions

A list of subscriptions.

type: array

items: Subscription (p. 518)

Subscription

Information about a subscription.

type: object

required: ["Id", "Source", "Subject", "Target"]

Id

The id of the subscription.

type: string
Source

The source of the subscription. Can be a thing ARN, a Lambda function ARN, 'cloud' (which represents the AWS IoT cloud), or 'GGShadowService'.

type: string

Subject

The subject of the message.

type: string

Target

Where the message is sent to. Can be a thing ARN, a Lambda function ARN, 'cloud' (which represents the AWS IoT cloud), or 'GGShadowService'.

type: string

Group

{
  "AssociatedAt": "string"
}

Group

type: object

AssociatedAt

The time, in milliseconds since the epoch, when the role ARN was associated with the group.

type: string

Group Certificate Authority Properties

{
  "GroupCertificateAuthorityId": "string",
  "GroupCertificateAuthorityArn": "string"
}

Group Certificate Authority Properties

Information about a certificate authority for a group.

type: object

GroupCertificateAuthorityId

The ID of the certificate authority for the group.

type: string

GroupCertificateAuthorityArn

The ARN of the certificate authority for the group.

type: string
GroupCertificateConfiguration

```json
{
    "GroupId": "string",
    "CertificateAuthorityExpiryInMilliseconds": "string",
    "CertificateExpiryInMilliseconds": "string"
}
```

Information about a group certificate configuration.

**GroupId**

The ID of the group certificate configuration.

*type: string*

**CertificateAuthorityExpiryInMilliseconds**

The amount of time remaining before the certificate authority expires, in milliseconds.

*type: string*

**CertificateExpiryInMilliseconds**

The amount of time remaining before the certificate expires, in milliseconds.

*type: string*

GroupInformation

```json
{
    "Name": "string",
    "Id": "string",
    "Arn": "string",
    "LastUpdatedTimestamp": "string",
    "CreationTimestamp": "string",
    "LatestVersion": "string",
    "LatestVersionArn": "string"
}
```

Information about a group.

**Name**

The name of the group.

*type: string*

**Id**

The ID of the group.

*type: string*
Arn
   The ARN of the group.
   type: string

LastUpdatedTimestamp
   The time, in milliseconds since the epoch, when the group was last updated.
   type: string

CreationTimestamp
   The time, in milliseconds since the epoch, when the group was created.
   type: string

LatestVersion
   The latest version of the group.
   type: string

LatestVersionArn
   The ARN of the latest version of the group.
   type: string

**GroupOwnerSetting**

```json
{
   "AutoAddGroupOwner": true,
   "GroupOwner": "string"
}
```

**GroupOwnerSetting**

Group owner related settings for local resources.
   type: object

**AutoAddGroupOwner**

If true, AWS Greengrass automatically adds the specified Linux OS group owner of the resource to the Lambda process privileges. Thus the Lambda process will have the file access permissions of the added Linux group.
   type: boolean

**GroupOwner**

The name of the Linux OS group whose privileges will be added to the Lambda process. This field is optional.
   type: string

**GroupVersion**

```json
{
}
```
GroupVersion

Information about a group version.

type: object
CoreDefinitionVersionArn

The ARN of the core definition version for this group.

type: string
DeviceDefinitionVersionArn

The ARN of the device definition version for this group.

type: string
FunctionDefinitionVersionArn

The ARN of the function definition version for this group.

type: string
SubscriptionDefinitionVersionArn

The ARN of the subscription definition version for this group.

type: string
LoggerDefinitionVersionArn

The ARN of the logger definition version for this group.

type: string
ResourceDefinitionVersionArn

The resource definition version ARN for this group.

type: string

IDList

{ "Ids": [ "string" ] }
ListDefinitionsResponse

A list of definitions.

type: object

Definitions

Information about a definition.

type: array

items: DefinitionInformation (p. 463)

DefinitionInformation

Information about a definition.

type: object

Name

The name of the definition.

   type: string

Id

The ID of the definition.

   type: string

Arn

The ARN of the definition.

   type: string

LastUpdatedTimestamp

The time, in milliseconds since the epoch, when the definition was last updated.

   type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the definition was created.
ListDeploymentsRequest

```json
{
    "MaxResults": 0,
    "NextToken": "string"
}
```

ListDeploymentsRequest

- **type:** object
- **MaxResults**
  - The maximum number of results to be returned per request.
  - `in`: query
  - `type`: integer
  - `min`: 1
  - `max`: 250
- **NextToken**
  - The token to retrieve the next set of results.
  - `in`: query
  - `type`: string

ListDeploymentsResponse

```json
{
    "Deployments": [
        {
            "GroupArn": "string",
            "DeploymentId": "string",
            "DeploymentArn": "string",
            "DeploymentType": "NewDeployment|Redeployment|ResetDeployment|ForceResetDeployment",
            "CreatedAt": "string"
        }
    ]
}
```
ListGroupCertificateAuthoritiesResponse

```
{
  "GroupCertificateAuthorities": [
    {
      "GroupCertificateAuthorityId": "string",
      "GroupCertificateAuthorityArn": "string",
      "PrivateKey": "string"
    }
  ],
  "NextToken": "string"
}
```

ListDeploymentsResponse

type: object

Deployments

type: array

items: Deployment (p. 464)

Deployment

Information about a deployment.

type: object

GroupArn

The ARN of the group for this deployment.

type: string

DeploymentId

The ID of the deployment.

type: string

DeploymentArn

The ARN of the deployment.

type: string

DeploymentType

type: string

enum: ["NewDeployment", "Redeployment", "ResetDeployment", "ForceResetDeployment"]

CreatedAt

The time, in milliseconds since the epoch, when the deployment was created.

createdAt: string

NextToken

The token for the next set of results, or null if there are no additional results.

in: query

type: string

ListGroupCertificateAuthoritiesResponse
ListGroupsResponse

  type: object

Groups

  Information about a group.

  type: array

  items: GroupInformation (p. 495)
GroupInformation

Information about a group.

type: object

Name

The name of the group.

type: string

Id

The ID of the group.

type: string

Arn

The ARN of the group.

type: string

LastUpdatedTimestamp

The time, in milliseconds since the epoch, when the group was last updated.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the group was created.

type: string

LatestVersion

The latest version of the group.

type: string

LatestVersionArn

The ARN of the latest version of the group.

type: string

NextToken

The token for the next set of results, or null if there are no additional results.

type: string

ListVersionsResponse

```
{
  "Versions": [
    {
      "Arn": "string",
      "Id": "string",
      "Version": "string",
      "CreationTimestamp": "string"
    }
  ],
  "NextToken": "string"
}
```
LocalDeviceResourceData

{  
  "SourcePath": "string",
  "GroupOwnerSetting": {  
    "AutoAddGroupOwner": true,
    "GroupOwner": "string"
  }
}

LocalDeviceResourceData

Attributes that define a local device resource.
LocalVolumeResourceData

- **SourcePath**
  - The local absolute path of the device resource. The source path for a device resource can refer only to a character device or block device under `/dev`.
  - type: string

- **GroupOwnerSetting**
  - Group owner related settings for local resources.
  - type: object
    - **AutoAddGroupOwner**
      - If true, AWS Greengrass automatically adds the specified Linux OS group owner of the resource to the Lambda process privileges. Thus the Lambda process will have the file access permissions of the added Linux group.
      - type: boolean

    - **GroupOwner**
      - The name of the Linux OS group whose privileges will be added to the Lambda process. This field is optional.
      - type: string

---

LocalVolumeResourceData

```
{
  "SourcePath": "string",
  "DestinationPath": "string",
  "GroupOwnerSetting": {
    "AutoAddGroupOwner": true,
    "GroupOwner": "string"
  }
}
```

- **SourcePath**
  - The local absolute path of the volume resource on the host. The source path for a volume resource type cannot start with `/proc` or `/sys`.
  - type: string

- **DestinationPath**
  - The absolute local path of the resource inside the Lambda environment.
  - type: string

- **GroupOwnerSetting**
  - Group owner related settings for local resources.
type: object
AutoAddGroupOwner
If true, AWS Greengrass automatically adds the specified Linux OS group owner of the resource to the Lambda process privileges. Thus the Lambda process will have the file access permissions of the added Linux group.

type: boolean
GroupOwner
The name of the Linux OS group whose privileges will be added to the Lambda process. This field is optional.

type: string

**Logger**

```json
{
  "Id": "string",
  "Type": "FileSystem|AWSCloudWatch",
  "Component": "GreengrassSystem|Lambda",
  "Level": "DEBUG|INFO|WARN|ERROR|FATAL",
  "Space": 0
}
```

**Logger**

Information about a logger
type: object
required: ["Id", "Type", "Component", "Level"]

**Id**
The id of the logger.
type: string

**Type**
type: string
enum: ["FileSystem", "AWSCloudWatch"]

**Component**
type: string
enum: ["GreengrassSystem", "Lambda"]

**Level**
type: string
enum: ["DEBUG", "INFO", "WARN", "ERROR", "FATAL"]

**Space**
The amount of file space, in KB, to use if the local file system is used for logging purposes.
type: integer
**LoggerComponent**

```
{
}
```

type: string
enum: ["GreengrassSystem", "Lambda"]

**LoggerDefinitionVersion**

```
{
  "Loggers": [
    {
      "Id": "string",
      "Type": "FileSystem|AWSCloudWatch",
      "Component": "GreengrassSystem|Lambda",
      "Level": "DEBUG|INFO|WARN|ERROR|FATAL",
      "Space": 0
    }
  ]
}
```

LoggerDefinitionVersion
Information about a logger definition version.
type: object
Loggers
A list of loggers.
type: array
items: **Logger (p. 505)**
Logger
Information about a logger
type: object
required: ["Id", "Type", "Component", "Level"]
Id
The id of the logger.
type: string
Type
type: string
enum: ["FileSystem", "AWSCloudWatch"]
Component
type: string
enum: ["GreengrassSystem", "Lambda"]
Level

type: string
enum: ["DEBUG", "INFO", "WARN", "ERROR", "FATAL"]

Space
The amount of file space, in KB, to use if the local file system is used for logging purposes.
type: integer

LoggerLevel

{
}

type: string
enum: ["DEBUG", "INFO", "WARN", "ERROR", "FATAL"]

LoggerType

{
}

type: string
enum: ["FileSystem", "AWSCloudWatch"]

Permission

{
}

The type of permission a function has to access a resource.
type: string
enum: ["ro", "rw"]

ResetDeploymentsRequest

{
  "Force": true
}

ResetDeploymentsRequest
Information about a group reset request.
type: object

Force

If true, performs a best-effort only core reset.

type: boolean

Reset_deployments_response

{
  "DeploymentArn": "string",
  "DeploymentId": "string"
}

Reset_deployments_response

type: object

DeploymentArn

The ARN of the deployment.

type: string

Deployment_Id

The ID of the deployment.

type: string

Resource

{
  "Id": "string",
  "Name": "string",
  "ResourceDataContainer": {
    "-- NOTE:": "choose one of the following --",
    "LocalDeviceResourceData": {
      "SourcePath": "string",
      "GroupOwnerSetting": {
        "AutoAddGroupOwner": true,
        "GroupOwner": "string"
      }
    },
    "LocalVolumeResourceData": {
      "SourcePath": "string",
      "DestinationPath": "string",
      "GroupOwnerSetting": {
        "AutoAddGroupOwner": true,
        "GroupOwner": "string"
      }
    },
    "SageMakerMachineLearningModelResourceData": {
      "SageMakerJobArn": "string",
      "DestinationPath": "string"
    },
    "S3MachineLearningModelResourceData": {
      "S3Uri": "string",
      "DestinationPath": "string"
    }
  }
}
Resource

Information about a resource.

type: object

required: ["Id", "Name", "ResourceDataContainer"]

Id

The resource ID, used to refer to a resource in the Lambda function configuration. Max length is 128 characters with pattern \[a-zA-Z0-9:_-]+\]. This must be unique within a Greengrass group.

  type: string

Name

The descriptive resource name, which is displayed on the AWS Greengrass console. Max length 128 characters with pattern \[a-zA-Z0-9:_-]+\]. This must be unique within a Greengrass group.

  type: string

ResourceDataContainer

A container for resource data. The container takes only one of the following supported resource data types: LocalDeviceResourceData, LocalVolumeResourceData, SageMakerMachineLearningModelResourceData, S3MachineLearningModelResourceData.

  type: object

NOTE: oneOf

LocalDeviceResourceData

Attributes that define a local device resource.

  type: object

SourcePath

The local absolute path of the device resource. The source path for a device resource can refer only to a character device or block device under /dev.

  type: string

GroupOwnerSetting

Group owner related settings for local resources.

  type: object

AutoAddGroupOwner

If true, AWS Greengrass automatically adds the specified Linux OS group owner of the resource to the Lambda process privileges. Thus the Lambda process will have the file access permissions of the added Linux group.

  type: boolean

GroupOwner

The name of the Linux OS group whose privileges will be added to the Lambda process. This field is optional.
LocalVolumeResourceData

Attributes that define a local volume resource.

type: object

SourcePath

The local absolute path of the volume resource on the host. The source path for a volume resource type cannot start with `/proc` or `/sys`.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.

type: string

GroupOwnerSetting

Group owner related settings for local resources.

type: object

AutoAddGroupOwner

If true, AWS Greengrass automatically adds the specified Linux OS group owner of the resource to the Lambda process privileges. Thus the Lambda process will have the file access permissions of the added Linux group.

type: boolean

GroupOwner

The name of the Linux OS group whose privileges will be added to the Lambda process. This field is optional.

type: string

SageMakerMachineLearningModelResourceData

Attributes that define an Amazon SageMaker machine learning resource.

type: object

SageMakerJobArn

The ARN of the Amazon SageMaker training job that represents the source model.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.

type: string

S3MachineLearningModelResourceData

Attributes that define an Amazon S3 machine learning resource.

type: object

S3Uri

The URI of the source model in an Amazon S3 bucket. The model package must be in tar.gz or .zip format.
ResourceAccessPolicy

A policy used by the function to access a resource.

type: object

required: ["ResourceId"]

ResourceId

The ID of the resource. (This ID is assigned to the resource when you create the resource definition.)

type: string

Permission

The type of permission a function has to access a resource.

type: string

enum: ["ro", "rw"]

ResourceDataContainer

```json
{   "*-- NOTE:": "choose one of the following --*",   "LocalDeviceResourceData": {     "SourcePath": "string",     "GroupOwnerSetting": {       "AutoAddGroupOwner": true,       "GroupOwner": "string"     }   },   "LocalVolumeResourceData": {     "SourcePath": "string",     "DestinationPath": "string",     "GroupOwnerSetting": {       "AutoAddGroupOwner": true,       "GroupOwner": "string"     }   },   "SageMakerMachineLearningModelResourceData": {     "SageMakerJobArn": "string",     "DestinationPath": "string"   } }
```
"S3MachineLearningModelResourceData": {
  "S3Uri": "string",
  "DestinationPath": "string"
}
}

A container for resource data. The container takes only one of the following supported resource data types: LocalDeviceResourceData, LocalVolumeResourceData, SageMakerMachineLearningModelResourceData, S3MachineLearningModelResourceData.

type: object

NOTE: oneOf

LocalDeviceResourceData

Attributes that define a local device resource.

type: object

SourcePath

The local absolute path of the device resource. The source path for a device resource can refer only to a character device or block device under /dev.

type: string

GroupOwnerSetting

Group owner related settings for local resources.

type: object

AutoAddGroupOwner

If true, AWS Greengrass automatically adds the specified Linux OS group owner of the resource to the Lambda process privileges. Thus the Lambda process will have the file access permissions of the added Linux group.

type: boolean

GroupOwner

The name of the Linux OS group whose privileges will be added to the Lambda process. This field is optional.

type: string

LocalVolumeResourceData

Attributes that define a local volume resource.

type: object

SourcePath

The local absolute path of the volume resource on the host. The source path for a volume resource type cannot start with /proc or /sys.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.
GroupOwnerSetting

Group owner related settings for local resources.

type: object

AutoAddGroupOwner

If true, AWS Greengrass automatically adds the specified Linux OS group owner of the resource to the Lambda process privileges. Thus the Lambda process will have the file access permissions of the added Linux group.

type: boolean

GroupOwner

The name of the Linux OS group whose privileges will be added to the Lambda process. This field is optional.

type: string

SageMakerMachineLearningModelResourceData

Attributes that define an Amazon SageMaker machine learning resource.

type: object

SageMakerJobArn

The ARN of the Amazon SageMaker training job that represents the source model.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.

type: string

S3MachineLearningModelResourceData

Attributes that define an Amazon S3 machine learning resource.

type: object

S3Uri

The URI of the source model in an Amazon S3 bucket. The model package must be in tar.gz or .zip format.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.

type: string

ResourceDefinitionVersion

```json
{
  "Resources": [
    {
      "Id": "string",
```
"Name": "string",
"ResourceDataContainer": {  
  "#-- NOTE:": "choose one of the following --*",
  "LocalDeviceResourceData": {  
    "SourcePath": "string",
    "GroupOwnerSetting": {  
      "AutoAddGroupOwner": true,
      "GroupOwner": "string"
    }
  },
  "LocalVolumeResourceData": {  
    "SourcePath": "string",
    "DestinationPath": "string",
    "GroupOwnerSetting": {  
      "AutoAddGroupOwner": true,
      "GroupOwner": "string"
    }
  },
  "SageMakerMachineLearningModelResourceData": {  
    "SageMakerJobArn": "string",
    "DestinationPath": "string"
  },
  "S3MachineLearningModelResourceData": {  
    "S3Uri": "string",
    "DestinationPath": "string"
  }
}  
}  

ResourceDefinitionVersion

Information about a resource definition version.

  type: object
Resources

A list of resources.

  type: array

  items: Resource (p. 508)

Resource

Information about a resource.

  type: object

  required: ["Id", "Name", "ResourceDataContainer"]

Id

The resource ID, used to refer to a resource in the Lambda function configuration. Max length is 128 characters with pattern [*a-zA-Z0-9_:\-]+. This must be unique within a Greengrass group.

  type: string

Name

The descriptive resource name, which is displayed on the AWS Greengrass console. Max length 128 characters with pattern [*a-zA-Z0-9_:\-]+. This must be unique within a Greengrass group.

  type: string
ResourceDataContainer

A container for resource data. The container takes only one of the following supported resource data types: LocalDeviceResourceData, LocalVolumeResourceData, SageMakerMachineLearningModelResourceData, S3MachineLearningModelResourceData.

- type: object
- NOTE: oneOf

LocalDeviceResourceData

Attributes that define a local device resource.

- type: object
- SourcePath
  
  The local absolute path of the device resource. The source path for a device resource can refer only to a character device or block device under /dev.

  - type: string
- GroupOwnerSetting
  
  Group owner related settings for local resources.

  - type: object
- AutoAddGroupOwner
  
  If true, AWS Greengrass automatically adds the specified Linux OS group owner of the resource to the Lambda process privileges. Thus the Lambda process will have the file access permissions of the added Linux group.

  - type: boolean
- GroupOwner
  
  The name of the Linux OS group whose privileges will be added to the Lambda process. This field is optional.

  - type: string

LocalVolumeResourceData

Attributes that define a local volume resource.

- type: object
- SourcePath
  
  The local absolute path of the volume resource on the host. The source path for a volume resource type cannot start with /proc or /sys.

  - type: string
- DestinationPath
  
  The absolute local path of the resource inside the Lambda environment.

  - type: string
- GroupOwnerSetting
  
  Group owner related settings for local resources.
type: object
AutoAddGroupOwner

    If true, AWS Greengrass automatically adds the specified Linux OS group owner of the resource to
    the Lambda process privileges. Thus the Lambda process will have the file access permissions of the
    added Linux group.

type: boolean
GroupOwner

    The name of the Linux OS group whose privileges will be added to the Lambda process. This field is
    optional.

type: string
SageMakerMachineLearningModelResourceData

    Attributes that define an Amazon SageMaker machine learning resource.

type: object
SageMakerJobArn

    The ARN of the Amazon SageMaker training job that represents the source model.

type: string
DestinationPath

    The absolute local path of the resource inside the Lambda environment.

type: string
S3MachineLearningModelResourceData

    Attributes that define an Amazon S3 machine learning resource.

type: object
S3Uri

    The URI of the source model in an Amazon S3 bucket. The model package must be in tar.gz or .zip
    format.

type: string
DestinationPath

    The absolute local path of the resource inside the Lambda environment.

type: string

S3MachineLearningModelResourceData

    Attributes that define an Amazon S3 machine learning resource.

    ```
    {  
        "S3Uri": "string",
        "DestinationPath": "string"
    }
    ```
type: object

S3Uri

The URI of the source model in an Amazon S3 bucket. The model package must be in tar.gz or .zip format.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.

type: string

S3UrlSignerRole

{ }

The IAM Role that Greengrass will use to create pre-signed URLs pointing towards the update artifact.

type: string

SageMakerMachineLearningModelResourceData

{ "SageMakerJobArn": "string", "DestinationPath": "string" } SageMakerMachineLearningModelResourceData

Attributes that define an Amazon SageMaker machine learning resource.

type: object

SageMakerJobArn

The ARN of the Amazon SageMaker training job that represents the source model.

type: string

DestinationPath

The absolute local path of the resource inside the Lambda environment.

type: string

SoftwareToUpdate

{ }


The piece of software on the Greengrass core that will be updated.

type: string
enum: ['core', 'ota_agent']

Subscription

```
{
  "Id": "string",
  "Source": "string",
  "Subject": "string",
  "Target": "string"
}
```

Subscription

Information about a subscription.

type: object

required: ['Id', 'Source', 'Subject', 'Target']

Id

The id of the subscription.

type: string

Source

The source of the subscription. Can be a thing ARN, a Lambda function ARN, 'cloud' (which represents the AWS IoT cloud), or 'GGShadowService'.

type: string

Subject

The subject of the message.

type: string

Target

Where the message is sent to. Can be a thing ARN, a Lambda function ARN, 'cloud' (which represents the AWS IoT cloud), or 'GGShadowService'.

type: string

SubscriptionDefinitionVersion

```
{
  "Subscriptions": [
    {
      "Id": "string",
      "Source": "string",
      "Subject": "string",
      "Target": "string"
    }
  ]
}
```
SubscriptionDefinitionVersion

Information about a subscription definition version.

type: object

Subscriptions

A list of subscriptions.

type: array

items: Subscription (p. 518)

Subscription

Information about a subscription.

type: object

required: ["Id", "Source", "Subject", "Target"]

Id

The id of the subscription.

type: string

Source

The source of the subscription. Can be a thing ARN, a Lambda function ARN, 'cloud' (which represents the AWS IoT cloud), or 'GGShadowService'.

type: string

Subject

The subject of the message.

type: string

Target

Where the message is sent to. Can be a thing ARN, a Lambda function ARN, 'cloud' (which represents the AWS IoT cloud), or 'GGShadowService'.

type: string

UpdateAgentLogLevel

The minimum level of log statements that should be logged by the OTA Agent during an update.

type: string

enum: ["NONE", "TRACE", "DEBUG", "VERBOSE", "INFO", "WARN", "ERROR", "FATAL"]
**UpdateConnectivityInfoRequest**

```json
{
  "ConnectivityInfo": [
    {
      "Id": "string",
      "HostAddress": "string",
      "PortNumber": 0x01,
      "Metadata": "string"
    }
  ]
}
```

Information required to update a Greengrass core's connectivity.

**ConnectivityInfo**

A list of connectivity info.

- type: array
- items: ConnectivityInfo (p. 458)

**Id**

The ID of the connectivity information.

- type: string

**HostAddress**

The endpoint for the Greengrass core. Can be an IP address or DNS.

- type: string

**PortNumber**

The port of the Greengrass core. Usually 8883.

- type: integer
  - format: int32

**Metadata**

Metadata for this endpoint.

- type: string

**UpdateConnectivityInfoResponse**

```json
{
  "message": "string",
  "Version": "string"
}
```
message
  A message about the connectivity info update request.
  type: string
Version
  The new version of the connectivity info.
  type: string

Update Targets

{}  

The ARNs of the targets (IoT things or IoT thing groups) that this update will be applied to.
  type: array

Update Targets Architecture

{}  

The architecture of the cores which are the targets of an update.
  type: string
  enum: ["armv7l", "x86_64", "aarch64"]

Update Targets Operating System

{}  

The operating system of the cores which are the targets of an update.
  type: string
  enum: ["ubuntu", "raspbian", "amazon_linux"]

Version Information

{  
  "Arn": "string",
  "Id": "string",
  "Version": "string",
}
"CreationTimestamp": "string"
}

VersionInformation

Information about a version.

type: object

Arn

The ARN of the version.

type: string

Id

The ID of the version.

type: string

Version

The unique ID of the version.

type: string

CreationTimestamp

The time, in milliseconds since the epoch, when the version was created.

type: string

Versions

{
  "Versions": [
    {
      "Arn": "string",
      "Id": "string",
      "Version": "string",
      "Version": "string",
      "CreationTimestamp": "string"
    }
  ]
}

Versions

type: object

Versions

A list of versions.

type: array

items: VersionInformation (p. 521)

VersionInformation

Information about a version.

type: object
Arn
   The ARN of the version.
   type: string
Id
   The ID of the version.
   type: string
Version
   The unique ID of the version.
   type: string
CreationTimestamp
   The time, in milliseconds since the epoch, when the version was created.
   type: string

updateGroupCertificateConfigurationRequest

```json
{
   "CertificateExpiryInMilliseconds": "string"
}
```

updateGroupCertificateConfigurationRequest
   type: object
   required: ["CertificateExpiryInMilliseconds"]
CertificateExpiryInMilliseconds
   The amount of time remaining before the certificate expires, in milliseconds.
   type: string