CLI reference for Edge Microgateway

Edge Microgateway v. 2.3.x

Overview

The Edge Microgateway CLI lets you control and manage all aspects of an Edge Microgateway instance.

Note:

The Setting up and configuring Edge Microgateway (https://docs.apigee.com/api-platform/microgateway/2.3.x/setting-and-configuring-edge-microgateway-v2.3.x.html) provides a good overview of the most commonly used commands.

Managing certificates

The `cert` commands let you install and manage the public/private key pair that is used to sign bearer tokens used by clients to make secure calls through Edge Microgateway. The keys are stored on Apigee Edge in a secure vault. Edge Microgateway uses the public key to validate signed bearer tokens. These keys are generated when you run the edgemicro configure command, as explained in Setting up and configuring Edge Microgateway (https://docs.apigee.com/api-platform/microgateway/2.3.x/setting-and-configuring-edge-microgateway-v2.3.x.html). You can use the **cert** commands described here to regenerate and manage these keys if needed.

Install a certificate

Installs the keys in a vault using the Apigee Edge secure store service and returns the public key as output. Key pairs are scoped to a specified organization.

Usage

```
edgemicro cert install -o [organization] -e [environment] -u [username] -p
```

Parameters

Parameters	Description
`-o, --org`	The Apigee organization for which you configured Edge Microgateway.
`-e, --env`	An environment for which you configured Edge Microgateway.
`-u, --username`	Your Apigee username. You must be an org administrator for the specified organization.
`-p, --password`	(Optional) Your password. You will be prompted if you do not provide this parameter on the command line.
`-f, --force`	(Optional) If a key is already stored in the vault, force its replacement.
`-h, --help`	Output usage information.

Example

```
edgemicro cert install -o docs -e test -u jdoe@example.com -f
```

Output

```
current nodejs version is v6.1.0
current edgemicro version is 2.2.4-beta
password:
deleting vault
creating vault
adding private_key
adding public_key
installed cert
```

Delete a certificate

Deletes the key pair for an organization.

Usage

```
edgemicro cert delete -o [organization] -e [environment] -u [username] -p
```

Parameters

Parameters	Description
`-o, --org`	The Apigee organization for which you configured Edge Microgateway.
`-e, --env`	An environment for which you configured Edge Microgateway.
`-u, --username`	Your Apigee username. You must be an org administrator for the specified organization.
`-p, --password`	(Optional) Your password. You will be prompted if you do not provide this parameter on the command line.
`-h, --help`	Output usage information.

Example

```
edgemicro cert delete -o docs -e test -u jdoe@example.com
```

Output

```
deleting vault
Vault deleted!
```

Check a certificate

Checks that your organization has a certificate installed.

Usage

```
edgemicro cert check -o [organization] -e [environment] -u [username] -p [
```

Parameters

Parameters	Description
`-o, --org`	The Apigee organization for which you configured Edge Microgateway.

`-e, --env`	An environment for which you configured Edge Microgateway.
`-u, --username`	Your Apigee username. You must be an org administrator for the specified organization.
`-p, --password`	(Optional) Your password. You will be prompted if you do not provide this parameter on the command line.
`-h, --help`	Output usage information.

Example

```
edgemicro cert check -o docs -e test -u jdoe@example.com
```

Output (Success)

```
checked cert successfully
```

Output (Failure)

If the certificate does not exist, an error is returned.

Get the public key

Returns the public key for the specified organization. Does not require authentication.

Usage

```
edgemicro cert public-key -o [organization] -e [environment]
```

Parameters

Parameters	Description
`-o, --org`	The Apigee organization for which you configured Edge Microgateway.
`-e, --env`	An environment for which you configured Edge Microgateway.
`-h, --help`	Output usage information.

Example

```
edgemicro cert public-key -o docs -e test
```

Output (Sample)

```
-----BEGIN CERTIFICATE-----

MIICpDCCAYwCCQCKpXWGum9uTjANBgkq9w0BAQsFADAUMRIwEAYDVQQDEwls
b2NhbGhvc3cNMTYxMTAyMjAxNTA2WhcNMTYxMTAzMjAxNTA2WjAUMRIwEAYD
VQQDEwlsb2Nvc3QwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQDD
ETr/ne/gP47+9LgjLvBJjpbFVdaeUabZZ3wbA7sSIVnnNXWt3yPklrWSfIh+
L2+dq4k+YunsQE2+dwPdXA2x8DPGvqGcCdWPxnpZ7ix5Akbs8i/J+Ue0pXT4
jkpWbaDDftGL2tcxwP45yI+slpDYvmWRps07TFXkAPjGEHYPvCT9/v/35GkL
0h4v0S+XfpcjS5c47o7CIWlkgEM5GBosQUb17kuVR48392vGFPsnCP34iNe7
zguhiCXYg2zkOcj/N3AE4xKIhCz0QcewULy08GltWMmyjJ/30zs8P41JqoA4
RwfeEZ8RenN3rJQim1ppAAEwDQYJKoZIhvcNAQELBQADggEBAHcORIfc+ySe
2FMmqicNe6Wt5V/5zOaBMLsGQdqGOHB5cQc79sMBrk680KVhrwHXZ2nBIkVe
UEL+2qLY1VLfswBNAvcNwV9U4TwHq4eIANiD751oJK1tHmM/ujThQkwNf96o
6I7Ws+wfpGO3ppJCttRbtnATIxrwsCHN4i5lbW/tJSL7t/Zf6T1B+YSJU9AF
xuvLq22cCqyYJQdmKq2vVS55WRZdllm+mRtJrv7MLM9gfDPRxVlwrBz/eQHy
Fv+dwzxcvQjkz03RNhZUodzuD34DPJoYeK++rItsddwQ26KTahH80yYTAFzY
x9gfPf1/+qo=?

-----END CERTIFICATE-----
```

Configuring Edge Microgateway for Apigee Edge Cloud

Enables Edge Microgateway to operate with an Apigee Edge Cloud instance. It wraps and performs a sequence of commands to deploy a required authentication proxy to Edge, generate authentication tokens, and update config files. For a complete working example, see the Setting up and configuring Edge Microgateway (https://docs.apigee.com/api-platform/microgateway/2.3.x/setting-and-configuring-edge-microgateway-v2.3.x.html)

.

Note:

Apigee recommends that you use the configure command to configure Edge Microgateway. This command is the only command needed to configure Edge Microgateway. It automates commands such as genkeys, deploy, update config, verify config, and so on.

Usage

```
edgemicro configure -o [organization] -e [environment] -u [username] -p [pa
```

Parameters

Parameters	Description
`-o, --org`	The Apigee organization for which you configured Edge Microgateway.
`-e, --env`	An environment for which you configured Edge Microgateway.
`-u, --username`	Your Apigee username. You must be an org administrator for the specified organization.
`-p, --password`	(Optional) Your password. You will be prompted if you do not provide this parameter on the command line.
`-r, --url`	(Optional) Specifies the virtual host alias for your organization:environment. See the Edge documentation to learn about virtual hosts. Default: `org-env.apigee.net`
`-v, --virtualHosts`	(Optional) Overrides the default virtual hosts, which are "`default,secure`". Use this option if you have virtual hosts specified for your Edge organization:environment other than these defaults. See the Edge documentation to learn about virtual hosts.
`-d, --debug`	(Optional) Provides debug information.
`-h, --help`	Output usage information.

Usage notes

This command stores configuration information in `~/.edgemicro/org-env-config.yaml`.

Example

```
edgemicro configure -o docs -e test -u jdoe@example.com
```

Output

Upon success, the command returns a set of keys that you need to use when you start Edge Microgateway.

```
...

The following credentials are required to start edge micro
  key: d2f0a246ad52b5d2a8b04ba65b43c76348aba586691cf6185cd7bb9fb78fe9f
  secret: 59813bc1db4a7ada182705ae40893c28a6fae680c3deb42aefbf1a4db42e372

edgemicro configuration complete!
```

Generating keys

The **genkeys** command generates a key and secret pair used by Edge Microgateway to authenticate itself when asynchronously posting analytics data to Apigee Edge.

Usage

```
edgemicro genkeys -o [organization] -e [environment] -u [username] -p [pas
```

Parameters

Parameters	Description
-o, --org	The Apigee organization for which you configured Edge Microgateway.
-e, --env	An environment for which you configured Edge Microgateway.
-u, --username	Your Apigee username. You must be an org administrator for the specified organization.
-p, --password	(Optional) Your password. You will be prompted if you do not provide this parameter on the command line.
-h, --help	Output usage information.

Usage notes

Upon success, the command returns three items. The first is a URL that you need to place in the configuration file. The other two are a key pair that are required when you start an Edge Microgateway instance.

- **bootstrap URL**: This URL points to an Apigee Edge service that enables an Edge Microgateway instance to send analytics data to Apigee Edge. You need to copy that URL into the Edge Microgateway config file: `~/.edgemicro/org-env-config.yaml`.
- **key**: The key. This key, and the secret, are required as input to the CLI command used to start an instance of Edge Microgateway.
- **secret**: The secret.

Example

```
edgemicro genkeys -o docs -e test -u jdoe@example.com
```

Output (Sample)

Upon success, the command returns a set of keys that you need to use when you start Edge Microgateway.

```
configuring host edgemicroservices-us-east-1.apigee.net for region us-east
Please copy the following property to the edge micro agent config
  bootstrap: https://edgemicroservices-us-east-1.apigee.net/edgemicro/bootstrap/

The following credentials are required to start edge micro
  key: db39324077989c75eac34c13d285772ea8e3b982b957b3f52916f3048511443f
  secret: 5bf8da39de0056f88fdd5f25a8602d22f312c1c8c870580a5fef55ac6637b7ae

finished
```

Configuring Edge Microgateway for Apigee Edge Private Cloud

Configures Edge Microgateway to work with an Apigee Edge Private Cloud installation.

Usage

```
edgemicro private configure -o [organization] -e [environment] -u [usernam
```

Parameters

Parameters	Description
`-o, --org`	The Apigee organization for which you configured Edge Microgateway.
`-e, --env`	An environment for which you configured Edge Microgateway.
`-u, --username`	Your Apigee username. You must be an org administrator for the specified organization.
`-p, --password`	(Optional) Your password. You will be prompted if you do not provide this parameter on the command line.
`-r, --runtime-url`	Specifies the runtime URL for your private cloud instance.
`-m, --mgmt-url`	The URL of the management server for your private cloud instance.
`-v, --virtualHosts`	Comma-separated list of virtual hosts for your organization:environment. Default "`default,secure`".
`-d, --debug`	(Optional) Provides debug information.
`-h, --help`	Output usage information.

Usage notes

This command stores configuration information in `~/.edgemicro/org-env-config.yaml`.

Note:

By default, the **edgemicro-auth** proxy expects to connect through a virtual host called **secure**. If your Private Cloud installation does not have this virtual host defined, you will receive a configuration error. Be sure to specify on the command line a **virtual_host** that exists. It is generally safe to specify a virtual host called **default**.

Example

```
edgemicro private configure -o docs -e test -u jdoe@example.com -r http://
```

Output

Upon success, the command returns a set of keys that you need to use when you start Edge Microgateway.

```
...

The following credentials are required to start edge micro
  key: d2f0a246ad52b5d2a8b04ba65b43c76348aba586691cf6185cd7bb9fb78fe9f
  secret: 59813bc1db4a7ada182705ae40893c28a6fae680c3deb42aefbf1a4db42e372

edgemicro configuration complete!
```

Starting Edge Microgateway

Before starting Edge Microgateway, you must first run the **edgemicro configure** (Public Cloud) or **edgemicro private configure** (Private Cloud). The configure command returns the key and secret values that are required to start Edge Microgateway.

Usage

```
edgemicro start -o [organization] -e [environment] -k [public-key] -s [sec
```

Parameters

Parameters	Description
`-o, --org`	The Apigee organization for which you configured Edge Microgateway.
`-e, --env`	An environment for which you configured Edge Microgateway.
`-k, --key`	The key value returned that is returned when you run the "**`edgemicro configure`**" command.
`-s, --secret`	The secret value returned that is returned when you run the "**`edgemicro configure`**" command.
`-p, --processes`	(Optional) The number of processes to start. Default: The number of cores on your system.
`-d, --pluginDir`	(Optional) Absolute path to the plugin directory.

`-r, --port`	(Optional) Overrides the port number specified in the ~**`/.edgemicro/org-env-config.yaml`** file. Default: 8000
`-c, --cluster`	(Optional) Starts Edge Microgateway in cluster mode. **Note: As of v2.3.1, this option has been removed. In v2.3.1 and later versions, Edge Micro always starts in cluster mode.**
`-c --config`	(Optional) Specifies the location of the **`default config.yaml`** file. By default, this file is in **`./config/config.yaml`**.
`-d, --debug`	(Optional) Provides debug information.
`-h, --help`	Output usage information.

Setting the port

The `start` command lets you specify a port number to override the port specified in the configuration file. You can also specify a port number using the `PORT` environment variable. For example:

```
edgemicro start -o docs -e test -k abc123 -s xyz456 -p 2 --port 8002
```

or

```
export PORT=8002
edgemicro start -o org -e test -k key -s secret -p 2
```

If the port is in use, Edge Microgateway returns an error.

About clustering

Edge Microgateway employs the Node.js cluster module (https://nodejs.org/api/cluster.html) to enable clustering. Clustering allows Edge Microgateway to take advantage of multi-core systems. For details, see this Node.js documentation (https://nodejs.org/api/cluster.html).

Note:

In v2.2.4-beta and later versions, Edge Microgateway always starts in cluster mode. If you are using an older version, you can start the microgateway in cluster mode with the `--cluster` option. For example:

```
edgemicro start -o docs -e test -k abc123 -s xyz456 --cluster
```

Example

```
edgemicro start -o docs -e test -k abc123 -s xyz456
```

Sample output:

```
...

PROCESS PID : 54709
installed plugin from analytics
installed plugin from analytics
installed plugin from oauth
installed plugin from oauth
installed plugin from analytics
installed plugin from oauth
5a86b570-a142-11e6-aa1f-6730e9065d6c edge micro listening on port 8000
5a86dc80-a142-11e6-962c-43d9cc723190 edge micro listening on port 8000
5a8751b0-a142-11e6-8241-cf1c517c91eb edge micro listening on port 8000
installed plugin from analytics
installed plugin from oauth
5a924e30-a142-11e6-8740-2944162ce275 edge micro listening on port 8000
```

Managing tokens

The `token` commands let you obtain, decode, and verify signed OAuth2 access tokens. See also Secure API calls with an OAuth2 access token (https://docs.apigee.com/microgateway/v2.3.x/setting-and-configuring-edge-microgateway-v2.3.x#Oauthtoken)
.

Decode a token

Decodes a signed, encoded bearer token into its plain-text JSON JWT (Java Web Token) representation. A token conveys information about the Apigee Edge developer app that provided the keys used to create the token, including application name, client_id, product list, and more.

Usage

```
edgemicro token decode -f [filename]
```

Parameters

Parameters	Description
`-f, --file`	The name of a file containing the JWT token to decode.
`-h, --help`	Output usage information.

Example

```
edgemicro token decode -f token.jwt
```

Output (Sample)

```
{ header: { typ: 'JWT', alg: 'RS256' },

  payload:
   { application_name: 'b43342ef-86f6-4666-a121-b9ac2025d217',
     client_id: 'O9ZQRZKnn1rdgcKQgsABSMdOsKS',
     scopes: [],
     api_product_list: [ 'MicroTest' ],
     iat: 1436280566,
     exp: 1436282365 },
  signature: '' }
```

Generate a token

Generates a signed bearer token. The token allows client apps to make authenticated API calls to Edge Microgateway. The token is an OAuth 2.0-compliant JSON Web Token (JWT). It requires as input the Consumer Key (client id) and Consumer Secret (client secret) values from a registered developer app on Apigee Edge. See also Secure API calls with an OAuth2 access token

(https://docs.apigee.com/microgateway/v2.3.x/setting-and-configuring-edge-microgateway-v2.3.x#Oauthtoken)

.

Usage

```
edgemicro token get -o [org] -e [env] -i [client_id] -s [client_secret]
```

Parameters

Parameters	Description
`-o, --org`	The Apigee organization for which you configured Edge Microgateway.
`-e, --env`	An environment for which you configured Edge Microgateway.
`-1, --key`	The Client ID from the Developer App associated with your Microgateway-aware proxy.
`-s, --secret`	The Client Secret from the Developer app associated with your Microgateway-aware proxy.
`-h, --help`	Output usage information.

Example

```
edgemicro token get -o docs -e test -i 5UzOwAXGoOeo60aew94PPG5MAZE3aJp -s
```

Output (Sample)

```
{ token: 'eyJ0eXAiOiJKV1JhbGciOiJSUzI1NiJ9Glvbl9uYW1lIjoiNWNiMGY0NTV6TV3
EtOWMzOC00YmJjLWIzNzEtZGMxZTQzOGMxIiwiY2xpZW50X2lkIjoiNVV6T3dBWEdvSU9lbz
YwYWV3OTRQN0c1TUFaRTNhSnAiLCJzY2MiOltdLCJhcGlfcHJvZHVjdF9saXN0IjpbIkVkZ2
VNaWNyb1Rlc3RQcm9kdWN0Il0sImlhdCI3ODEyMzQ2MSwiZXhwIjoxNDc4MTI1MjYwfQ.Dx5
f5U7PXm8koNGmFX4N6VrxKMJnpndKgoJ5zWSJvBZ6Ccvhlpd85ipIIA5S2A5nx4obYWp_rpY
RJpIGYwyxP6Oq2j0rxnVjdCC4qyYMgthZjhKgEBVBe3s1ndP72GP2vV6PsSA9RQ2-yzsy9r0
TzhAZ3NJTxT1tS0XKqKngE-0hR3fJHVLAzdMDT0AmS9H0Z2NAJtQOuK6RTpCjG9B6Bc48AEM
sj7QSM-1LWiQ8LdY8k_BoC06qsTI7bCQGWwTuqL-ismbcx2bxovUxSemZIaoROfuF-dCZHG3
2aTP75WxBvvNgBBvPvQtPzbeSOtEaww' }
```

Making an HTTP request to get a token

You can also make a raw HTTP request to get the token. Here's a curl example. Just substitute your org and environment names in the URL, and substitute the Consumer Id and Consumer Secret values for the client_id and client_secret params:

```
curl -i -X POST "http://<org>-<test>.apigee.net/edgemicro-auth/token" -d '
```

Verify a token

Verifies a signed bearer token against the public key stored on Apigee Edge for the specified organization and environment.

Usage

```
edgemicro token verify -o [org] -e [env] -f [filename]
```

Parameters

Parameters	Description
`-o, --org`	The Apigee organization for which you configured Edge Microgateway.
`-e, --env`	An environment for which you configured Edge Microgateway.
`-f, --file`	The name of a file containing the JWT to verify.
`-h, --help`	Output usage information.

Example

```
edgemicro token get -o docs -e test -f token.jwt
```

Sample output for valid token

```
{ application_name: 'b43342ef-86f6-4666-a121-b9ac2025d217',

  client_id: 'O9ZQRZKnn1rdgcKQsAZUBkQSMdOsKS',

  scopes: [],

  api_product_list: [ 'MicroTest' ],
  iat: 1436396155,

  exp: 1436397954 }
```

Sample output for invalid token

```
{ [JsonWebTokenError: invalid token] name: 'JsonWebTokenError', message: '
```

Sample output for expired token

```
{ [TokenExpiredError: jwt expired]

  name: 'TokenExpiredError',

  message: 'jwt expired',

  expiredAt: Tue Jul 07 2015 09:19:25 GMT-0600 (MDT) }
```

Obtaining bearer tokens directly

You may prefer to obtain bearer tokens directly, by making an HTTP request to the token endpoint on Apigee Edge. The actual token endpoint is implemented in the proxy that is deployed with the deploy-edge-service CLI command.

Here's a curl example. Just substitute your org and environment names in the URL, and substitute the Consumer Id and Consumer Secret values obtained from a developer app on Apigee Edge for the **client_id** and **client_secret** params:

```
curl -i -X POST "http://<org>-<test>.apigee.net/edgemicro-auth/token" -d '
```

Sample output:

```
HTTP/1.1 200 OK

X-Powered-By: Express

Cache-Control: no-store
```

```
Pragma: no-cache

Content-Type: application/json; charset=utf-8

Content-Length: 640

ETag: W/"280-ze/g/k+c9taqp110vjYQ"

Date: Fri, 17 07 2015 15:49:24 GMT

Connection: keep-alive

"<long string of numbers and letters>"
```

Initializing a new Edge Microgateway configuration

Run this command once after you first install Edge Microgateway. Creates a new default configuration file: `~/.edgemicro/default.yaml`.

Usage

```
edgemicro init
```

Parameters

Parameters	Description
`-h, --help`	Output usage information.

Example

```
edgemicro init
```

Output (Success)

```
config initialized to /MyHome/.edgemicro/default.yaml
```

Verifying Edge Microgateway configuration

Verifies that Edge Microgateway is properly configured.

Usage

```
edgemicro verify -o [organization] -e [environment] -k [public-key] -s [se
```

Parameters

Parameters	Description
`-o, --org`	The Apigee organization for which you configured Edge Microgateway.
`-e, --env`	An environment for which you configured Edge Microgateway.
`-k, --key`	The key value returned that is returned when you run the "**`edgemicro configure`**" command.
`-s, --secret`	The secret value returned that is returned when you run the "**`edgemicro configure`**" command.
`-h, --help`	Output usage information.

Example

```
edgemicro verify -o docs -e test -k abc123 -s xyz456
```

Output (Success)

```
logging to /var/tmp/edgemicro-My-Machine.local-a0c48610-a148-11e6-8466-93f
installed plugin from analytics
installed plugin from oauth
a0c48610-a148-11e6-8466-93f081b05988 edge micro listening on port 8000
verifying analytics negative case: OK
verifying bootstrap url availability:OK
verifying jwt_public_key availability: OK
verifying products availability: OK
verifying quota with configured products: OK
verifying analytics with payload: OK
verification complete
```

Check the microgateway cluster status

Added: v2.2.4-beta

By default, Edge Microgateway starts in cluster mode. You can use this command to check the status of the cluster.

Note:

You must run this command from the same directory where you started Edge Microgateway. This command requires the presence of an `edgemicro.sock` file. You can check for the presence of this file by running: `ls -rtl`.

Usage

```
edgemicro status
```

Parameters

Parameters	Description
`-h, --help`	Output usage information.

Example

```
edgemicro status
```

Output (Success)

```
current nodejs version is v6.1.0
current edgemicro version is 2.2.4-beta
edgemicro is running with 4 workers
```

Stopping the microgateway cluster

Added: v2.2.4-beta

Stops the Edge Microgateway cluster.

Note:

You must run this command from the same directory where you started Edge Microgateway. This command requires the presence of an edgemicro.sock file. You can check for the presence of this file by running: `ls -rtl`.

Usage

```
edgemicro stop
```

Parameters

Parameters	Description
`-h, --help`	Output usage information.

Example

```
edgemicro stop
```

Output (Success)

```
current nodejs version is v6.1.0
current edgemicro version is 2.2.4-beta
Stop Completed Successfully
```

Reloading the microgateway cluster

Added: v2.2.4-beta

Provides zero-downtime restart after a configuration change. Reloads the Edge Microgateway by pulling in a new configuration.

Note:

You must run this command from the same directory where you started Edge Microgateway. This command requires the presence of an edgemicro.sock file. You can check for the presence of this file by running: `ls -rtl`.

Usage

```
edgemicro reload -o [organization] -e [environment] -k [public-key] -s [se
```

Parameters

>

Parameters	Description
`-o, --org`	The Apigee organization for which you configured Edge Microgateway.
`-e, --env`	An environment for which you configured Edge Microgateway.
`-k, --key`	The key value returned that is returned when you run the "**`edgemicro configure`**" command.
`-s, --secret`	The secret value returned that is returned when you run the "**`edgemicro configure`**" command.
`-h, --help`	Output usage information.

Example

```
edgemicro reload -o docs -e test -k abc123 -s xyz456
```

Output (Success)

```
...

Reload Completed Successfully
```

Last updated March 7, 2018.