EVERTRUST

API Gateway

2025-10-24

Table of Contents

Chapter 1. Overview

Horizon API Gateway is a reverse proxy built on top of the Caddy webserver. It offers all Caddy capabilities while providing a built-in plugin `horizon-api-gateway`, to validate and filter SCEP requests to Horizon.

Chapter 2. Install Horizon API Gateway

RPM

Horizon API Gateway is available as a packaged RPM. This RPM will install the following components:

- the `horizon-api-gateway` binary in `/usr/bin`
- the configuration file in `/etc/horizon-api-gateway/Caddyfile`
- the service file in `/usr/lib/systemd/system/horizon-api-gateway.service`
- the `horizon-api-gateway` user and group that the service runs as

It will also start the service with the default configuration, running on port 80.

Installation from the EverTrust repository

Create a `/etc/yum.repos.d/horizon-api-gateway.repo` file containing the EverTrust repository info:

```
[horizon]
enabled=1
name=Horizon API Gateway Repository
baseurl=https://repo.evertrust.io/repository/horizon-api-gateway-rpm/
gpgcheck=0
username=<username>
password=<password>
```

Replace `<username>` and `<password>` with the credentials you were provided.

You can then run the following to install the latest Horizon version:

```
$ yum install horizon-api-gateway
```

To prevent unattended upgrades when running yum update, you should pin the Horizon version by adding

```
exclude=horizon-api-gateway
```

at the end of the `/etc/yum.repos.d/horizon-api-gateway.repo` file after installing Horizon API Gateway.

Installing from RPM

Upload the file `horizon-api-gateway_<version>.rpm` through SCP under `/root`.

Access the server through SSH with an account with administrative privileges;

Install the Horizon API Gateway package with the following command:

```
$ yum localinstall /root/horizon-api-gateway_<version>.rpm
```

Docker

Horizon API Gateway is available as a docker image. This image expects:

- the configuration file mounted in `/etc/Caddyfile`

Without this configuration, it will fail to start

Chapter 3. Configuration

Horizon API Gateway is a Caddy webserver and offers all configuration capabilities of this webserver.

The configuration detailed here is of the `horizon_api_gateway` plugin, as well as an example configuration.

The configuration file is installed in `/etc/horizon-api-gateway/Caddyfile` in RPM and should be mounted in `/etc/Caddyfile`.

Plugin configuration

The following options are available

Parameter	Mandatory	Description	Example
endpoint	☑	The Horizon endpoint to validate the SCEP payloads against	horizon-url.com
allow_invalid		Forward SCEP requests that do not pass validation to the next handler	-
soft_fail		If an internal issue occurs while validation the SCEP request (failure to retrieve the SCEP RA from Horizon), forward the validation to the next handler	-
capabilities		Define which type of requests to forward to the next handler (others will be blocked) in the list: `scep`, `intune`, `jamf`, `trustchain`, `all`	`scep`, `trustchain`
api_id		Credentials to authenticate on horizon	scep-proxy-username
api_key		Credentials to authenticate on horizon	scep-proxy-password

Parameter	Mandatory	Description	Example
cert_file		Path to client certificate to authenticate on horizon	/etc/horizon-api-gateway/cert.pem
key_file		Path to private key of client certificate to authenticate on horizon	/etc/horizon-api-gateway/key.pem
client_certificate_header		Name of header to populate with client certificate for certificate authentication.	SSL_CLIENT_CERT

Authentication

In order to retrieve SCEP RAs for SCEP validation, an authorization is required on Horizon.

The authorization is: **Protocols › SCEP › Proxy**.

Once the authorization is set, it can be given to a local account or a certificate.

Local account

When using a local account, two parameters must be set on the proxy:

- `api_id` the username
- `api_key` the password

Certificate

When using a certificate, two parameters must be set on the proxy:

- `cert_file` the path to the certificate
- `key_file` the path to the key

Capabilities

When enabling capabilities on the gateway, it allows blocking unwanted requests.

When no capabilities are defined, all requests are allowed

- `scep`: Allows requests for SCEP protocol profiles in Horizon
- `intune`: Allows requests for Intune profiles in Horizon
- `jamf`: Allows requests for Jamf profiles in Horizon

- `trustchain`: Allows requests for trust chain requests in Horizon
- `all`: Allows all other types of requests

For example, for an Intune deployment, the following capabilities configuration should be set:

```
capabilities intune
```

To use the gateway with the Horizon CLI in SCEP mode, the following capabilities configuration should be set:

```
capabilities scep,trustchain
```

Error handling

Multiple parameters allow fine grained control over SCEP validation errors:

- `soft_fail`: When an internal error occurs on the Proxy (failure to retrieve the SCEP RA, ...), making it so that the payload cannot be validated, forward it to the next handler nonetheless. This is false by default, so a failure to retrieve the SCEP Ra will result in a blocked request.
- `allow_invalid`: When no internal error occurs on the Proxy but the request cannot be validated,forward it to the next handler nonetheless. This is false by default, so an invalid request will result in a blocked request.

Reference

Plugin configuration should look like this in the `Caddyfile`:

```
horizon-api-gateway <endpoint> {
    api_id <api_id>
    api_key <api_key>
    cert_file <cert_file>
    key_file <key_file>
    client_certificate_header <certificate header name>
    capabilities <capabilities, comma separated>
    allow_invalid
    soft_fail
}
```

Example

To use the horizon-api-gateway in the following conditions:

- Listening on `intune-proxy.com` (https)
- Horizon instance is located at `horizon.com` (https)

- Using a certificate to authenticate on Horizon
- Should be used for Intune only

```
# Listening on intune-proxy.com:443
intune-proxy.com {
  # Exposing server certificate
  tls /path/to/cert/for/intune-proxy.com /path/to/key/for/intune-proxy.com

  # Configuring the horizon_api_gateway plugin
  horizon_api_gateway horizon.com {
    cert_file /path/to/cert/for/horizon-authentication.pem
    key_file /path/to/cert/for/horizon-authentication.pem
    capabilities intune
    # Configuring trust_pool for custom CAs if needed (see
https://caddyserver.com/docs/caddyfile/directives/reverse_proxy#tls_trust_pool)
    tls_trust_pool inline {
        trust_der <der_encoded_certificate>
    }
  }

  # Once requests are allowed, proxy them to horizon.com
  reverse_proxy https://horizon.com {
    # Required if using SNI
    header_up Host horizon.com
    # Configuring trust_pool for custom CAs if needed (see
https://caddyserver.com/docs/caddyfile/directives/reverse_proxy#tls_trust_pool)
    tls_trust_pool inline {
        trust_der <der_encoded_certificate>
    }
  }
}
```