EVERTRUST

EverTrust Horizon documentation v2.3

Administration Guide

EVERTRUST

Table of Contents

1. Introduction

1.1. Description

Horizon is EverTrust's Certificate lifecycle management solution and is powered up by:

- Akka
- BouncyCastle
- MongoDB
- Kamon
- Play! Framework
- Scala
- NGINX
- Vue.js
- Quasar

This document is specific to Horizon version **2.3**.

1.2. Scope

This document is an administration guide detailing how to configure and operate Horizon.

1.3. Out of Scope

This document does not describe how to install and bootstrap a Horizon instance. Please refer to the installation guide for installation related tasks.

2. User Information

This is the section where to find all your profile information (identifier, email, name, authentication type, role and permissions), your preferences and change your account password (local account authentication only).

2.1. Profile access

1. Log in to Horizon.

2. Access your profile from the header by clicking on your account name.

2.2. How to change your password

1. Profile access

2. Fill your local password and confirm it.

3. Click on the 'Change Password' button.

CAUTION	Changing your password is only available if you are using a local account.

2.3. How to change your preferences

1. Profile access

2. Change your preferences:

- Appearance (light/dark mode)
- Horizon default language

3. Click on the 'Save' button.

3. Certification Authorities

This section details how to configure the Certification Authorities known by EverTrust Horizon.

3.1. Prerequisites

Certification Authorities will be needed beforehand, in one of these formats:

- Certificate file (PEM or DER).
- Certificate string (PEM).

You might also need the URL of the CRL issued by the CA, and/or the URL of the OCSP Responder for that CA.

3.2. How to configure a Certification Authority

1. Log in to Horizon Administration Interface.

2. Access Certification Authorities from the drawer or card: **Certification Authorities**.

3. Click on  .

Certificate Tab:

4. Either

- Fill in the certificate section with certificate string (PEM) OR
- Import the certificate file (PEM or DER).

Then click on the next button.

Details Tab:

5. Check the information from your CA certificate. Then click on the next button.

Configuration Tab:

6. Fill in the information you want to add.

- **Name*** *(string input)* :
  Enter a meaningful certificate authority name. It must be unique for each certificate authority.
- **OCSP responder URL** *(string)*:
  URL to request an OCSP responder.
- **CRL URL** *(string)*:
  URL to download the CA CRL.
- **Refresh Period** *(finite duration)*:
  CRL or OCSP Refresh Period. Must be a valid finite duration.

- **Timeout** *(finite duration)*:
  Connection timeout when reaching CRL or OCSP. Must be a valid finite duration.
- Proxy *(option)*:
  The HTTP/HTTPS proxy to use to reach the CRL or the OCSP Responder, if any.
- **Is trusted for server authentication** *(boolean)*:
  Tells whether the CA should be trusted for server authentication, aka SSL/TLS server trust. The default value is set to false.
- **Is trusted for client authentication** *(boolean)*:
  Tells whether the CA should be trusted for client authentication. The default value is set to false.
- **Outdated Revocation Status Policy** *(option)*:
  Select "Revoked" if you want all certificates to be handled as revoked if the CRL/OCSP are unavailable. Select "Last available status" if you want Horizon to use the last available revocation status for the certificates.

7. Click on the import button.

You can edit  , download  or delete  the Certification Authorities.

CAUTION	You will not be able to delete a Certification Authority if it is referenced in any other configuration element. Pay also attention that the CA might be used (e.g. for TLS trust chain building), even if it is not explicitly referenced in configuration items.

4. PKIs

4.1. PKI Queue

This section details how to configure a PKI Queue. PKI Queue are used to limit the PKI requests (enrollment, revocation)

4.1.1. PKI Queue Configuration

1. Log in to Horizon Administration Interface.

2. Access PKI Queues from the drawer or card: **PKI** > **PKI Queues**.

3. Click on  .

4. Fill in the fields:

- **Name*** *(string input)*:
  Choose a meaningful queue name. It must be unique.
- **Description** *(string input)*:
  The description for the PKI Queue.
- **Throttle Parallelism** *(int input)*:
  Number of requests processed at the same time.
- **Throttle Duration** *(finite duration)*:
  Maximum requests processed at the same time in a given duration. Parallelism must be set.
- **Max Size*** *(int input)*:
  Maximum requests stored in the queue
- **Cluster Wide** *(boolean)*:
  If not enabled, then the `throttleParallelism` and `throttleDuration` will be the same for all nodes in the cluster. If enabled, then the `throttleParallelism` and `throttleDuration` is generalized for all clusters.

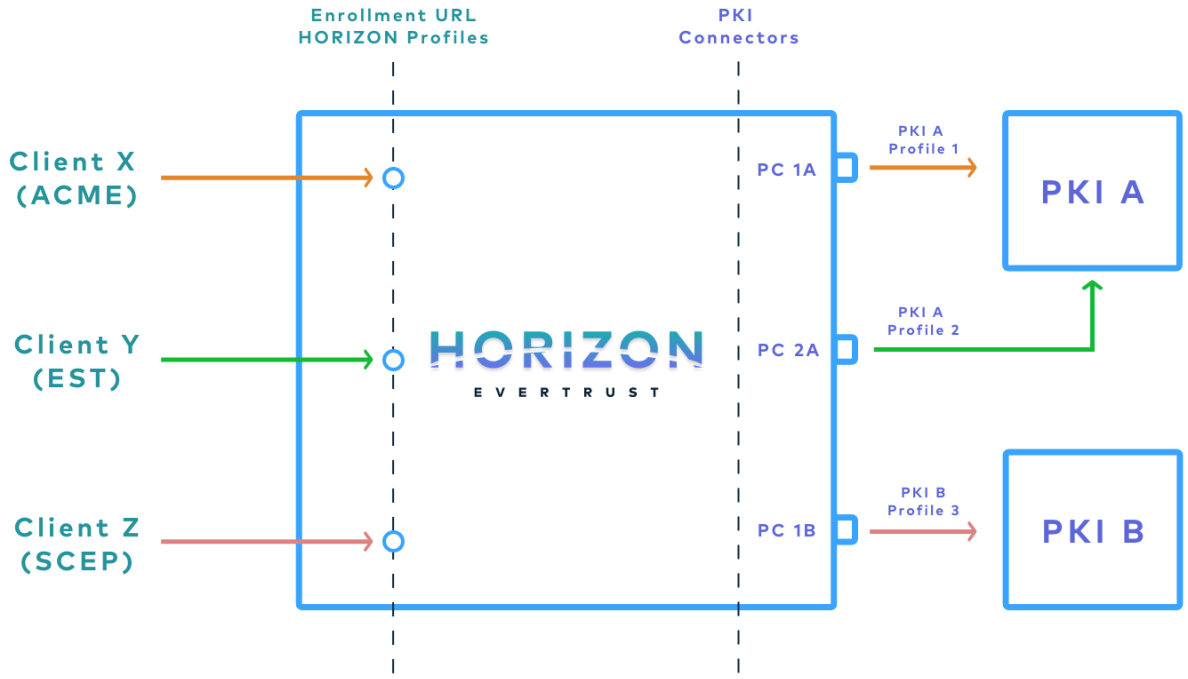
CAUTION	If the queue is full every new request will be discarded.

4.2. PKI Connectors

4.2.1. PKI Connectors

Description

A "PKI Connector" is a configuration piece that allows to establish the communication with any supported PKI. Additionally, it enables to map a specific certificate profile within the connected PKI.



Common prerequisites

To grant "Horizon" proper access to a given PKI, three categories of requirements must be gathered:

- **Credentials** : It could be either certificates (PKCS#12 format) or technical accounts (login/password) allowing to authenticate against the PKI API.
- **Permissions** : The credentials must be granted with the proper permissions on the PKI in order to be able to manage certificate lifecycle (enroll, revoke, renew).
- **Profile/Certificate information** : This information is used to map certificate types and/or certificate fields.

4.2.2. AWS PKI

Prerequisites

- You need to create a user using AWS IAM, and give it the `AWSCertificateManagerPrivateCAUser` right.
- You need to retrieve the Private CA ARN from ACM Private CA console.

NOTE	Refer to the editor's documentation to configure the PKI side here.

Create the PKI connector

1. Log in to Horizon Administration Interface.

2. Access PKI from the drawer or card: **PKI** > **PKI Connectors**.

3. Click on  .

4. Select the correct PKI type.

5. Click on the next button

General tab

6. Fill in the common mandatory fields:

- **Connector Name*** *(string input)*:
  Choose a meaningful connector name allowing to identify the mapping between the PKI and the Certificate Profile. It must be unique and must not contain spaces.
- **Proxy** *(string select)*:
  If the PKI is not directly reachable from Horizon, you can set up an HTTP/HTTPS proxy to properly forward the traffic.
- **Queue PKI** *(string select)*:
  The PKI Queue used to manage the PKI Requests (enrollment, revocation).
- **Timeout** *(finite duration)*:
  Represents a predefined interval of time without a PKI response, when the time has passed "Horizon" will cease trying to establish the communication. Must be in valid finite duration.

7. Click on the next button

Details tab

8. Fill in all mandatory fields:

- **AWS Region*** *(string input)*:
  AWS region to use.
- **AWS PCA ARN*** *(string input)*:
  Amazon Resource Name (ARN) is a file naming convention used to identify a particular resource in AWS public cloud. To be retrieved from AWS ACM Console.
- **AWS PCA Template ARN** *(string input)*:
  A template is a declaration of the AWS resources that make up a stack. The default value is set to: `arn:aws:acm-pca:::template/EndEntityCertificate/V1`.
- **AWS PCA Role ARN** *(string input)*
- **Certificate Policy OID** *(string input)*:
  An identifying number, in the form of an "object identifier" that is included in the `certificatePolicies` field of a certificate.
- **Certificate signing hash** *(string multiple)*:
  Select the hash function that will be used.
- **Certificate Usage** *(string multiple)*:
  Select the certificate usage.
- **Number of valid days** *(finite duration)*:

Certificate validity duration in days. Must be in valid finite duration. The default value is set to 365 days.

- **Retry Interval** *(finite duration)*:
  Predefined interval of time before retrying to retrieve the certificate from AWS. Must be in valid finite duration. The default value is set to 3 seconds.

9. Click on the next button

Authentication tab

10. Fill in the PKI-authentication fields:

- **AWS user access key ID** *(string input)*:
  Find AWS Account and Access Keys.
- **AWS user secret key** *(string input)*:
  Must be set only if and only if AWS user access key ID is set.

11. Click on the save button.

You can edit  , duplicate  or delete  the AWS PKI connector.

4.2.3. CertEurope PKI

Prerequisites

- A technical account should be created.
- This technical account must have permissions to enroll and revoke SSL certificates on the desired domain(s).

Limitations

- Only the following fields are managed: `commonName` and `subjectAltName DNS`.
- For multi-valued fields (SAN DNS), if more data items are provided than configured in CCS for the given "Offer Identifier", the exceeding items will be ignored.
- All limitations induced by the use of the CCS REST Connector.

Create the PKI connector

1. Log in to Horizon Administration Interface.

2. Access PKI from the drawer or card: **PKI** > **PKI Connectors**.

3. Click on  .

4. Select the correct PKI type.

5. Click on the next button

General tab

6. Fill in the common mandatory fields:

- **Connector Name*** *(string input)*:
  Choose a meaningful connector name allowing to identify the mapping between the PKI and the Certificate Profile. It must be unique and must not contain spaces.
- **Proxy** *(string select)*:
  If the PKI is not directly reachable from Horizon, you can set up an HTTP/HTTPS proxy to properly forward the traffic.
- **Queue PKI** *(string select)*:
  The PKI Queue used to manage the PKI Requests (enrollment, revocation).
- **Timeout** *(finite duration)*:
  Represents a predefined interval of time without a PKI response, when the time has passed "Horizon" will cease trying to establish the communication. Must be in valid finite duration.

7. Click on the next button

Details tab

8. Fill in all mandatory fields:

- **Endpoint URL to the CSS partner API*** *(string input)*:
  URL to access the CertEurope web service API.
- **Technical account login*** *(string input)*:
  Login of the technical account created in CCS, usually an email address.
- **Technical account password*** *(string input)*:
  Password of the technical account created in CCS.
- **CCS offer identifier*** *(string input)*:
  The identifier of the offer within CCS.
- **Organization ID*** *(string input)*:
  Customer organization ID. For French companies, it's usually the "SIREN".
- **Revocation reason** *(string select)*:
  Select from the drop down the default revocation reason.
- **Interval before retrying to retrieve certificate** *(finite duration)*:
  The default value is set to 21 seconds.

9. Click on the next button.

Authentication tab

10. Fill in the PKI-authentication fields:

- **Authentication PKCS#12*** *(import)*:
  PKCS#12 file containing the authentication certificate used to connect to the PKI.
- **PKCS#12 Password*** *(string input)*: Enter the password used to secure the aforementioned

PKCS#12.

11. Click on the save button.

You can edit  , duplicate  or delete  the CertEurope PKI connector.

4.2.4. CS-Novidy's TrustyKey PKI

Prerequisites

- A technical account should be created.
- This technical account must have permissions to enroll and revoke SSL certificates on the desired certificate profiles.
- An authentication and a signature certificate must be issued under as PKCS#12 files for this account.

Limitations

- Only the following fields are managed: `commonName (as mail_lastname)`, `contactEmail (as mail_email)`, `OU (as org_unit)`, `O (as corp_company)`, `C (as country)`, `UID (as employeeID)`, `subjectAltNames DNS` and `msUPN`.
- For multi-valued fields (SAN DNS), if more data items are provided than configured in TrustyKey for the given `PGC`, the exceeding items will be ignored.
- All limitations induced by the use of the TrustyKey CMP Connector.

Create the PKI connector

1. Log in to Horizon Administration Interface.

2. Access PKI from the drawer or card: **PKI** > **PKI Connectors**.

3. Click on  .

4. Select the correct PKI type.

5. Click on the next button

General tab

6. Fill in the common mandatory fields:

- **Connector Name*** *(string input)*:
  Choose a meaningful connector name allowing to identify the mapping between the PKI and the Certificate Profile. It must be unique and must not contain spaces.
- **Proxy** *(string select)*:
  If the PKI is not directly reachable from Horizon, you can set up an HTTP/HTTPS proxy to

properly forward the traffic.

- **Queue PKI** *(string select)*:
  The PKI Queue used to manage the PKI Requests (enrollment, revocation).
- **Timeout** *(finite duration)*:
  Represents a predefined interval of time without a PKI response, when the time has passed "Horizon" will cease trying to establish the communication. Must be in valid finite duration.

7. Click on the next button

Details tab

8. Fill in all mandatory fields:

- **API endpoint URL*** *(string input)*:
  URL to access the CS-Novidy's TrustyKey web service.
- **PGC*** *(string input)*:
  Enter name of the `PGC` to be used.
- **TrustyKey PKI server DN*** *(string input)*:
  Enter the DN of the TrustyKey PKI server, starting from the CN.
- **TrustyKey PKI server Certificate*** *(string input)*:
  Enter the PEM representing the certificate of the CA issuing the certificates.
- **CN mapping** *(string input)*:
  Enter a CN to be mapped.
- **Email mapping** *(string input)*: Enter an email address or domain to be mapped.
- **SAN DNS mapping** *(string input)*:
  Enter a SAN DNS to be mapped.
- **Profile mapping** *(string input)*:
  Enter a profile to be mapped.
- **Issuer mapping** *(string input)*:
  Enter an issuer to be mapped.
- **Legacy CMP Style** *(boolean)*
  Chose whether to use the legacy CMP style.

9. Click on the next button.

Authentication tab

10. Fill in the PKI-authentication fields:

- **Authentication PKCS#12*** *(import p12)*:
  Import the PKCS#12 file containing the authentication certificate used to connect to the PKI.
- **PKCS#12 Password*** *(string input)*:
  Enter the password used to secure the aforementioned PKCS#12.
- **Signer PKCS#12*** *(import p12)*:
  Import the PKCS#12 file containing the signature certificate used to sign the CMP messages.

- **Signer certificate password*** *(string input)*:
  Enter the password used to secure the aforementioned PKCS#12.

11. Click on the save button.

You can edit , duplicate or delete the CS-Novidy's TrustyKey PKI connector.

4.2.5. DigiCert CertCentral PKI

Prerequisites

- You need to validate the domain(s) for which you will issue certificates prior to their issuance. This can be done in DigiCert CertCentral in the Certificates > Domains menu.
- You need to retrieve the `organizationId` from DigiCert CertCentral in the Certificates > Organizations menu.
- You need to generate an API Key in DigiCert CertCentral using the Account > Account Access menu.

Limitations

- Only the following fields are managed: `commonName` and `subjectAltName DNS` and `RFC822Name`.
- For multi-valued fields (SAN DNS and `RFC822Name`), if more data items are provided than configured in DigiCert CertCentral for the given type of certificate, the exceeding items will be ignored.
- All limitations induced by the use of the DigiCert CertCentral REST Connector.

Create the PKI connector

1. Log in to Horizon Administration Interface.

2. Access PKI from the drawer or card: **PKI** > **PKI Connectors**.

3. Click on .

4. Select the correct PKI type.

5. Click on the next button

General tab

6. Fill in the common mandatory fields:

- **Connector Name*** *(string input)*:
  Choose a meaningful connector name allowing to identify the mapping between the PKI and the Certificate Profile. It must be unique and must not contain spaces.
- **Proxy** *(string select)*:

If the PKI is not directly reachable from Horizon, you can set up an HTTP/HTTPS proxy to properly forward the traffic.

- **Queue PKI** *(string select)*:
  The PKI Queue used to manage the PKI Requests (enrollment, revocation).
- **Timeout** *(finite duration)*:
  Represents a predefined interval of time without a PKI response, when the time has passed "Horizon" will cease trying to establish the communication. Must be in valid finite duration.

7. Click on the next button

Details tab

8. Fill in all mandatory fields:

- **DigiCert CertCentral API endpoint*** *(string input or select)*:
  URL to access DigiCert CertCentral API along with the certificate type to issue. To do so you can select from the drop down menu or type in your "certificate offer" value, then press "Enter" the corresponding URL will be automatically fetched.
- **DigiCert CertCentral Customer Organization ID*** *(int)*:
  Enter customer organization ID.
- **DigiCert CertCentral CA Cert ID** *(int)*:
  Enter CA Cert ID, to be used for private CA only.
- **Interval before retrying to retrieve certificate** *(finite duration)*:
  Use for private CA only. The default value is set to 9 seconds.
- **Skip Approval** *(boolean)*:
  The default value is set to false.

9. Click on the next button.

Custom tab

10. Click on  if custom data mapping is needed.

11. Fill in the PKI-custom data mapping:

- **Custom data field*** *(string input)*:
- **Label field*** *(select)*:
  Any existing Horizon Label

12. Click on the next button.

Authentication tab

13. Fill in the PKI-authentication fields:

- **DigiCert CertCentral API Key*** *(string input)*:
  Enter the API Key.

14. Click on the save button.

You can edit , duplicate or delete the DigiCert CertCentral PKI connector.

4.2.6. EJBCA PKI

Prerequisites

- A certificate profile should be created, e.g. reusing the default "SERVER" certificate profile.
- An authentication certificate should be issued for Horizon, and it should be given certificate issuance and revocation permissions on the aforementioned certificate procedure.

Limitations

- Only the following fields are managed: all Subject DN fields and `subjectAltNames DNS, IPaddress, RFC822Name, msUPN` and `msGUID`.
- For multi-valued fields (SAN DNS and `RFC822Name`), if more data items are provided than configured in EJBCA for the given *End Entity* profile, the exceeding items will be ignored.
- All limitations induced by the use of the EJBCA RA SOAP Connector.

Create the PKI connector

1. Log in to Horizon Administration Interface.

2. Access PKI from the drawer or card: **PKI** > **PKI Connectors**.

3. Click on .

4. Select the correct PKI type.

5. Click on the next button

General tab

6. Fill in the common mandatory fields:

- **Connector Name*** *(string input)*:
  Choose a meaningful connector name allowing to identify the mapping between the PKI and the Certificate Profile. It must be unique and must not contain spaces.
- **Proxy** *(string select)*:
  If the PKI is not directly reachable from Horizon, you can set up an HTTP/HTTPS proxy to properly forward the traffic.
- **Queue PKI** *(string select)*:
  The PKI Queue used to manage the PKI Requests (enrollment, revocation).
- **Timeout** *(finite duration)*:
  Represents a predefined interval of time without a PKI response, when the time has passed

"Horizon" will cease trying to establish the communication. Must be in valid finite duration.

7. Click on the next button

Details tab

8. Fill in all mandatory fields:

- **EJBCA RA URL*** *(string input)*:
  Enter SOAP endpoint URL of the EJBCA WebService.
- **EJBCA Certificate Profile Name*** *(string input)*:
  Enter EJBCA Certificate Profile to map for certificate issuance.
- **EJBCA CA Name*** *(string input)*:
  Enter CA to use for certificate issuance.
- **EJBCA End Entity Profile*** *(string input)*:
  Enter EJBCA End Entity profile.

9. Click on the next button.

Authentication tab

10. Fill in the PKI-authentication fields:

- **Authentication PKCS#12*** *(import p12)*:
  Import the PKCS#12 file containing the authentication certificate used to connect to the PKI.
- **PKCS#12 Password*** *(string input)*:
  Enter the password used to secure the aforementioned PKCS#12.

11. Click on the save button.

You can edit , duplicate or delete the EJBCA PKI connector.

4.2.7. Entrust Certificate Services PKI

Prerequisites

- A technical account should be created to be used with the API.
- This technical account must have permissions to enroll and revoke SSL certificates on the desired certificate profiles (superadmin role).

Limitations

- Only the following fields are managed: `commonName (as cn, for SMIME certs)`, `contactEmail (as requester email address)`, `OU (only one)` and `subjectAltName DNS (for SSL certs)` and `RFC822Name (for SMIME)`.
- For multi-valued fields (SAN DNS), if more data items are provided than configured in ECS for the given certificate type, the exceeding items will be ignored.

- All limitations induced by the use of the ECS REST Connector.

Create the PKI connector

1. Log in to Horizon Administration Interface.

2. Access PKI from the drawer or card: **PKI** > **PKI Connectors**.

3. Click on  .

4. Select the correct PKI type.

5. Click on the next button

General tab

6. Fill in the common mandatory fields:

- **Connector Name*** *(string input)*:
  Choose a meaningful connector name allowing to identify the mapping between the PKI and the Certificate Profile. It must be unique and must not contain spaces.
- **Proxy** *(string select)*:
  If the PKI is not directly reachable from Horizon, you can set up an HTTP/HTTPS proxy to properly forward the traffic.
- **Queue PKI** *(string select)*:
  The PKI Queue used to manage the PKI Requests (enrollment, revocation).
- **Timeout** *(finite duration)*:
  Represents a predefined interval of time without a PKI response, when the time has passed "Horizon" will cease trying to establish the communication. Must be in valid finite duration.

7. Click on the next button

Details tab

8. Fill in all mandatory fields:

- **Technical account API login*** *(string input)*:
  Enter the login of the technical account API.
- **Technical account API password*** *(string input)*:
  Enter the password of the technical account API.
- **Certificate Type** *(select)*:
  Select the Certificate Type to issue.
- **Requester's default email*** *(string input)*:
  Enter the requester default email address.
- **Requester's name** *(string input)*:
  Enter the requester name to register.
- **Requester's phone** *(string input)*:

Enter the requester phone to register.

- **Certificate lifetime** *(finite duration)*: Enter Certificate lifetime, in days. For `SMIME_ENT` it is the number of years. The default value is set to 90 days.
- **Client ID** *(int)*:
  Enter Client ID. The default value is set to 1.

9. Click on the next button.

Authentication tab

10. Fill in the PKI-authentication fields:

- **Authentication PKCS#12*** *(import p12)*:
  Import the PKCS#12 file containing the authentication certificate used to connect to the PKI.
- **PKCS#12 Password*** *(string input)*:
  Enter the password used to secure the aforementioned PKCS#12.

11. Click on the save button.

You can edit , duplicate or delete the Entrust Certificate Services PKI connector.

4.2.8. EverTrust integrated CA

Create the PKI connector

1. Log in to Horizon Administration Interface.

2. Access PKI from the drawer or card: **PKI** > **PKI Connectors**.

3. Click on .

4. Select the correct PKI type.

5. Click on the next button

General tab

6. Fill in the common mandatory fields:

- **Connector Name*** *(string input)*:
  Choose a meaningful connector name allowing to identify the mapping between the PKI and the Certificate Profile. It must be unique and must not contain spaces.
- **Proxy** *(string select)*:
  If the PKI is not directly reachable from Horizon, you can set up an HTTP/HTTPS proxy to properly forward the traffic.
- **Queue PKI** *(string select)*:
  The PKI Queue used to manage the PKI Requests (enrollment, revocation).

- **Timeout** *(finite duration)*:
  Represents a predefined interval of time without a PKI response, when the time has passed "Horizon" will cease trying to establish the communication. Must be in valid finite duration.

7. Click on the next button

Details tab

8. Fill in all mandatory fields:

- **Certificate Type*** *(select)*:
  Specify the certificate type to issue.
- **Signing algorithm*** *(select)*:
  Specify the signing algorithm.
- **CA Certificate** *(string input)*:
  Enter CA certificate.
- **CA Key** *(string input)*:
  Enter CA key.
- **CRL save path** *(string input)*:
  Path to save the CRL on the Horizon server.
- **CRL lifetime** *(finite duration)*:
  CRL lifetime in days. Must be a valid finite duration.
- **Certificate Back Date** *(finite duration)*:
  Certificate Back Date. Must be a valid duration.
- **Check Proof of Possession** *(boolean)*

9. Click on the save button.

You can edit  , duplicate  or delete  the EverTrust integrated CA PKI connector.

4.2.9. EverTrust Stream CA

Prerequisites

- A certificate template should be created in Stream for Horizon to enroll certificates upon.
- A dedicated Horizon account should be created in Stream and should have all lifecycle permissions on the desired CA. The credentials of this account should be either login and password or a PKCS#12 authentication certificate.

Create the PKI connector

1. Log in to Horizon Administration Interface.

2. Access PKI from the drawer or card: **PKI** > **PKI Connectors**.

3. Click on  .

4. Select the correct PKI type.

5. Click on the next button

General tab

6. Fill in the common mandatory fields:

- **Connector Name*** *(string input)*:
  Choose a meaningful connector name allowing to identify the mapping between the PKI and the Certificate Profile. It must be unique and must not contain spaces.
- **Proxy** *(string select)*:
  If the PKI is not directly reachable from Horizon, you can set up an HTTP/HTTPS proxy to properly forward the traffic.
- **Queue PKI** *(string select)*:
  The PKI Queue used to manage the PKI Requests (enrollment, revocation).
- **Timeout** *(finite duration)*:
  Represents a predefined interval of time without a PKI response, when the time has passed "Horizon" will cease trying to establish the communication. Must be in valid finite duration.

7. Click on the next button

8. Fill all mandatory fields:

- **Endpoint*** *(string input)*:
  Fill in the Stream endpoint url.
- **Template*** *(sting input)*:
  Fill in the Stream certificate template to enroll upon.
- **CA** *(string input)*:
  Fill in the Stream CA enrolling certificate (internal name).

9. Click on the next button.

Authentication tab

- **Authentication PKCS#12*** *(import p12)*:
  Import the PKCS#12 file containing the authentication certificate used to connect to the PKI.
- **PKCS#12 Password*** *(string input)*:
  Password used to secure the aforementioned PKCS#12.
- **Login*** *(string input)*:
  Enter the login for the dedicated Horizon account on Stream.
- **Password*** *(string input)*:
  Enter the aforementioned account's password .

10. Click on the save button.

You can edit , duplicate or delete the Evertrust Stream PKI connector.

4.2.10. FISId PKI

Create the PKI connector

1. Log in to Horizon Administration Interface.

2. Access PKI from the drawer or card: **PKI** > **PKI Connectors**.

3. Click on .

4. Select the correct PKI type.

5. Click on the next button

General tab

6. Fill in the common mandatory fields:

- **Connector Name*** *(string input)*:
  Choose a meaningful connector name allowing to identify the mapping between the PKI and the Certificate Profile. It must be unique and must not contain spaces.
- **Proxy** *(string select)*:
  If the PKI is not directly reachable from Horizon, you can set up an HTTP/HTTPS proxy to properly forward the traffic.
- **Queue PKI** *(string select)*:
  The PKI Queue used to manage the PKI Requests (enrollment, revocation).
- **Timeout** *(finite duration)*:
  Represents a predefined interval of time without a PKI response, when the time has passed "Horizon" will cease trying to establish the communication. Must be in valid finite duration.

7. Click on the next button

Details tab

8. Fill in all mandatory fields:

- **FISId endpoint URL*** *(string input)*:
  URL to access the API.
- **Template ID*** *(int)*:
  Enter the template ID.
- **Default owner ID*** *(string input)*:
  Enter a default owner ID.
- **Authentication domain ID*** *(int)*:
  Enter an authentication domain ID.

- **Owner groups** *(string input)*:
  Enter one or several, separated by commas
- **To delete after revocation** *(boolean)*:
  The default value is set to false.

9. Click on the next button.

Authentication tab

10. Fill in the PKI-authentication fields:

- **API Key*** *(string input)*:
  Enter app key.

11. Click on the save button.

You can edit  , duplicate  or delete  the FISId PKI connector.

4.2.11. GlobalSign Atlas PKI

Create the PKI connector

1. Log in to Horizon Administration Interface.

2. Access PKI from the drawer or card: **PKI** > **PKI Connectors**.

3. Click on  .

4. Select the correct PKI type.

5. Click on the next button

General tab

6. Fill in the common mandatory fields:

- **Connector Name*** *(string input)*:
  Choose a meaningful connector name allowing to identify the mapping between the PKI and the Certificate Profile. It must be unique and must not contain spaces.
- **Proxy** *(string select)*:
  If the PKI is not directly reachable from Horizon, you can set up an HTTP/HTTPS proxy to properly forward the traffic.
- **Queue PKI** *(string select)*:
  The PKI Queue used to manage the PKI Requests (enrollment, revocation).
- **Timeout** *(finite duration)*:
  Represents a predefined interval of time without a PKI response, when the time has passed "Horizon" will cease trying to establish the communication. Must be in valid finite duration.

7. Click on the next button

Details tab

8. Fill in all mandatory fields:

- **Hash Algorithm** *(select)*:
  Select the hash algorithm for the certificate to be issue.
- **API Key*** *(string input)*:
  Enter the key that allows to authenticate against GlobalSign Atlas API.
- **API Secret*** *(string input)*:
  Enter the password used to secure the aforementioned API Key.
- **Certificate Usage** *(select)*:
  Select a usage from the drop down list.
- **Retry Interval** *(finite duration)*:
  The default value is set to 3 seconds.

9. Click on the next button.

Authentication tab

10. Fill in the PKI-authentication fields:

- **Authentication PKCS#12*** *(import p12)*:
  Import the PKCS#12 file containing the authentication certificate used to connect to the PKI.
- **PKCS#12 Password*** *(string input)*:
  Enter the password used to secure the aforementioned PKCS#12.

10. Click on the save button.

You can edit , duplicate or delete the GlobalSign Atlas PKI connector.

4.2.12. GlobalSign MSSL PKI

Prerequisites

- A technical account should be created.
- This technical account must have permissions to enroll and revoke SSL certificates on the desired domain.

Limitations

- Only the following fields are managed: `contactEmail` and `subjectAltName DNS`.
- For multi-valued fields (SAN DNS), if more data items are provided than configured in GlobalSign MSSL for the given "Product", the exceeding items will be ignored.
- All limitations induced by the use of the GlobalSign MSSL SOAP Connector.

Create the PKI connector

1. Log in to Horizon Administration Interface.

2. Access PKI from the drawer or card: **PKI** > **PKI Connectors**.

3. Click on  .

4. Select the correct PKI type.

5. Click on the next button

General tab

6. Fill in the common mandatory fields:

- **Connector Name*** *(string input)*:
  Choose a meaningful connector name allowing to identify the mapping between the PKI and the Certificate Profile. It must be unique and must not contain spaces.
- **Proxy** *(string select)*:
  If the PKI is not directly reachable from Horizon, you can set up an HTTP/HTTPS proxy to properly forward the traffic.
- **Queue PKI** *(string select)*:
  The PKI Queue used to manage the PKI Requests (enrollment, revocation).
- **Timeout** *(finite duration)*:
  Represents a predefined interval of time without a PKI response, when the time has passed "Horizon" will cease trying to establish the communication. Must be in valid finite duration.

7. Click on the next button

Details tab

8. Fill in all mandatory fields:

- **GlobalSign endpoint*** *(string select)*:
  Select from the drop-down list: the value must be "prod" for GlobalSign Production endpoint or "test" for the test environment.
- **GlobalSign profile ID*** *(string input)*:
  To be retrieved from the URL in the GlobalSign MSSL console.
- **GlobalSign domain ID*** *(string input)*:
  The ID of the domain to manage. Displayed in the GlobalSign MSSL console.
- **Certificate validity** *(int input)*:
  Certificate validity in months.
- **Default email address** *(string input)*:
  Choose a default email address.
- **Default phone number** *(string input)*:
  Choose a default phone number.

- **Interval before retrying to retrieve certificate** *(finite duration)*:
  The default value is set to 9 seconds.

9. Click on the next button.

Authentication tab

10. Fill in the PKI-authentication fields:

- **Technical account username*** *(string input)*:
  Username of the technical account created in GlobalSign MSSL.
- **Technical account password*** *(string input)*:
  Password of the technical account created in GlobalSign MSSL.

11. Click on the save button.

You can edit  , duplicate  or delete  the GlobalSign MSSL PKI connector.

4.2.13. MetaPKI

Prerequisites

Endpoint issuing CA

Create the PKI connector

1. Log in to Horizon Administration Interface.

2. Access PKI from the drawer or card: **PKI** > **PKI Connectors**.

3. Click on  .

4. Select the correct PKI type.

5. Click on the next button

General tab

6. Fill in the common mandatory fields:

- **Connector Name*** *(string input)*:
  Choose a meaningful connector name allowing to identify the mapping between the PKI and the Certificate Profile. It must be unique and must not contain spaces.
- **Proxy** *(string select)*:
  If the PKI is not directly reachable from Horizon, you can set up an HTTP/HTTPS proxy to properly forward the traffic.
- **Queue PKI** *(string select)*:
  The PKI Queue used to manage the PKI Requests (enrollment, revocation).

- **Timeout** *(finite duration)*:
  Represents a predefined interval of time without a PKI response, when the time has passed "Horizon" will cease trying to establish the communication. Must be in valid finite duration.

7. Click on the next button

Details tab

8. Fill in all mandatory fields:

- **Endpoint*** *(string input)*:
  The MetaPKI Endpoint.
- **Endpoint Issuing CA*** *(string select)*:
  Select the CA that will be issuing the certificates for this connector (from the imported Horizon CAs)
- **Profile*** *(string input)*:
  Example: Applications_Auth_Client_Serveur_SSL.
- **Profile Cle*** *(string input)*:
  Example: Serveur_SSL
- **Workflow*** *(string input)*:
  Example: S_LOCAL_SOFT
- **Form Porteur Name** *(string input)*
- **Valid Days** *(finite duration)*
  Certificate lifetime in days (must be a valid finite duration).

9. Click on the next button.

Authentication tab

10. Fill in the PKI-authentication fields:

- **Authentication PKCS#12*** *(import p12)*:
  Import the PKCS#12 file containing the authentication certificate used to connect to the PKI.
- **PKCS#12 Password*** *(string input)*:
  Enter the password used to secure the aforementioned PKCS#12.

11. Click on the save button.

You can edit  , duplicate  or delete  the MetaPKI PKI connector.

4.2.14. Microsoft Active Directory Certificate Services PKI

Prerequisites

- The EverTrust ADCS Connector component must be installed

To install the *EverTrust ADCS Connector* component, follow these steps:

1. On a Windows Server 2016+ installed with ADCS, double-click on the EverTrust ADCS Connector MSI.
2. Follow the installation wizard.
3. If not already present, enroll a Web Server certificate for the server, so that it is available in the Local Machine store.
4. Edit the file located in `C:\Program Files\EverTrust\ADCSConnector\EverTrustADCSConnector.exe.config` and set the value of the key `CertHash` to the hash of the Web Server Certificate mentioned in the previous step.
5. Open the port 4443 on the firewall.
6. Start the service `EverTrust ADCS Connector`.

Finally, you must also create in the ADCS domain a technical user for Horizon, grant the appropriate permissions within ADCS to manage and issue and revoke certificates, and issue an Enrollment Agent certificate in PKCS#12 format.

- A technical account with appropriate permissions and an enrollment agent certificate must be created.

NOTE	It is possible to install the EverTrust ADCS Connector on another machine than an ADCS server. You then need to copy `C:\Windows\System32\certadm.dll` onto that server, and call `regsvr32 C:\Windows\System32\certadm.dll` from an elevated command prompt.

Limitations

- All limitations induced by the use of ADCS.

Create the PKI connector

1. Log in to Horizon Administration Interface.

2. Access PKI from the drawer or card: **PKI** > **PKI Connectors**.

3. Click on  .

4. Select the correct PKI type.

5. Click on the next button

General tab

6. Fill in the common mandatory fields:

- **Connector Name*** *(string input)*:
  Choose a meaningful connector name allowing to identify the mapping between the PKI and the Certificate Profile. It must be unique and must not contain spaces.
- **Proxy** *(string select)*:

If the PKI is not directly reachable from Horizon, you can set up an HTTP/HTTPS proxy to properly forward the traffic.

- **Queue PKI** *(string select)*:
  The PKI Queue used to manage the PKI Requests (enrollment, revocation).
- **Timeout** *(finite duration)*:
  Represents a predefined interval of time without a PKI response, when the time has passed "Horizon" will cease trying to establish the communication. Must be in valid finite duration.

7. Click on the next button

Details tab

8. Fill in all mandatory fields:

- **Endpoint*** *(string input)*:
  URL to access the Web Enrollment ADCS component.
- **Active Directory Domain Netbios Name*** *(string input)*:
  The Active Directory domain where to find the technical user and the ADCS server.
- **Profile*** *(string input)*:
  It is recommended to duplicate default template, and grant the enrolment permissions to the created technical user. Example: Web Server
- **CA Config*** *(string input)*:
  The `CaConfig` string, as given out by `certutil -config` for the considered ADCS CA. It's usually in the form `<ADCS Hostname>\<CA CommonName>`

9. Click on the next button.

Authentication tab

10. Fill in the PKI-authentication fields:

- **Signer PKCS#12*** *(import)*:
  Import the PKCS#12 file containing the signature certificate used to sign the CMP messages.
- **Sign certificate password*** *(string input)*:
  Enter the password used to secure the aforementioned PKCS#12.
- **Technical account MSUPN*** *(string input)*:
  Create a user that has certificate issuance, revocation and management permissions at CA level and for the relevant certificates template.
- **Technical account password*** *(string input)*:
  Password associated with aforementioned defined user.

11. Click on the save button.

You can edit  , duplicate  or delete  the Microsoft Active Directory Certificate Services PKI connector.

4.2.15. Nexus Certificate Manager PKI

Prerequisites

- A certificate procedure and a token procedure should be created.
- An authentication certificate should be issued for Horizon, and it should be given certificate issuance and revocation permissions on the aforementioned token procedure.
- Nexus Endpoint CA

Limitations

- Only the following fields are managed: `commonName, UID, OU, O, C` and `subjectAltNames DNS, IPaddress, RFC822Name` and `msUPN`.
- For multi-valued fields (SAN DNS, RFC822Name and IP address), if more data items are provided than configured in Nexus CM Procedure, the exceeding items will be ignored.
- All limitations induced by the use of the Nexus CM SDK.

Create the PKI connector

1. Log in to Horizon Administration Interface.

2. Access PKI from the drawer or card: **PKI** > **PKI Connectors**.

3. Click on  .

4. Select the correct PKI type.

5. Click on the next button

General tab

6. Fill in the common mandatory fields:

- **Connector Name*** *(string input)*:
  Choose a meaningful connector name allowing to identify the mapping between the PKI and the Certificate Profile. It must be unique and must not contain spaces.
- **Proxy** *(string select)*:
  If the PKI is not directly reachable from Horizon, you can set up an HTTP/HTTPS proxy to properly forward the traffic.
- **Queue PKI** *(string select)*:
  The PKI Queue used to manage the PKI Requests (enrollment, revocation).
- **Timeout** *(finite duration)*:
  Represents a predefined interval of time without a PKI response, when the time has passed "Horizon" will cease trying to establish the communication. Must be in valid finite duration.

7. Click on the next button

Details tab

8. Fill all mandatory fields:

- **Nexus CM DNS name*** *(string input)*:
  URL to access the Nexus Certificate Manager.
- **Nexus endpoint CA*** *(select)*:
  Select the endpoint CA.
- **Nexus CM Certificate procedure name*** *(string input)*:
  The token procedure name to use.
  Should point to the appropriate certificate procedure, and must be on PKCS#10 format.

9. Click on the next button.

Authentication tab

10. Fill in the PKI-authentication fields:

- **Authentication PKCS#12*** *(import p12)*:
  Import the PKCS#12 file containing the authentication certificate used to connect to the PKI.
- **PKCS#12 Password*** *(string input)*:
  Enter the password used to secure the aforementioned PKCS#12.

11. Click on the save button.

You can edit  , duplicate  or delete  the Nexus Certificate Manager PKI connector.

4.2.16. OpenTrust PKI

Prerequisites

- A certificate profile should be created.
- An authentication certificate should be issued for Horizon, and it should be given certificate issuance and revocation permissions on the aforementioned certificate profile.

Limitations

- Only the following fields are managed: `commonName, userID, serialNumber, organizationalUnit, organization, country, adminEmail` or `contactEmail, msCertTemplateName` and `subjectAltNames DNS, IPadress, RFC822Name, msUPN` and `msGUID`.
- For multi-valued fields (SAN DNS, IP address and RFC822Name), if more data items are provided than configured in OTPKI 'certificate template name', the exceeding items will be ignored.
- All limitations induced by the use of the RA SOAP Connector.

Create the PKI connector

1. Log in to Horizon Administration Interface.

2. Access PKI from the drawer or card: **PKI** > **PKI Connectors**.

3. Click on  .

4. Select the correct PKI type.

5. Click on the next button

General tab

6. Fill in the common mandatory fields:

- **Connector Name*** *(string input)*:
  Choose a meaningful connector name allowing to identify the mapping between the PKI and the Certificate Profile. It must be unique and must not contain spaces.
- **Proxy** *(string select)*:
  If the PKI is not directly reachable from Horizon, you can set up an HTTP/HTTPS proxy to properly forward the traffic.
- **Queue PKI** *(string select)*:
  The PKI Queue used to manage the PKI Requests (enrollment, revocation).
- **Timeout** *(finite duration)*:
  Represents a predefined interval of time without a PKI response, when the time has passed "Horizon" will cease trying to establish the communication. Must be in valid finite duration.

7. Click on the next button

Details tab

8. Fill in all mandatory fields:

- **OTPKI RA Connector URL*** *(string input)*:
  Must point to the "RA" connector URL.
- **OTPKI Certificate template name*** *(string input)*:
  The OTPKI certificate template to use.
- **OTPKI zone** *(string input)*:
  Specify a zone (if used).
- **Contact email mapping** *(string input)*:
  Allows to change the default fields names accordingly to certificate profiles.
- **SAN DNS mapping** *(string input)*:
  Allows to change the default fields names accordingly to certificate profiles.
- **SAN Email mapping** *(string input)*:
  Allows to change the default fields names accordingly to certificate profiles.

- **UID mapping** *(string input)*:
  Allows to change the default fields names accordingly to certificate profiles.

9. Click on the next button.

Authentication tab

10. Fill in the PKI-authentication fields:

- **Authentication PKCS#12*** *(import p12)*:
  Import the PKCS#12 file containing the authentication certificate used to connect to the PKI.
- **PKCS#12 Password*** *(string input)*:
  Enter the password used to secure the aforementioned PKCS#12.

11. Click on the save button.

You can edit , duplicate or delete the OpenTrust PKI connector.

4.2.17. Sectigo CMS PKI

Prerequisites

- For publicly trusted certificates, you need to validate the domain(s) for which you will issue certificates prior to their issuance.
- You need to retrieve the `customerUri` and the `organizationId` from Sectigo CMS.
- You need to create a technical account with appropriate permissions including the `allow ssl auto approve` permission. You need to set a password for the technical account.

Limitations

- Only the `subjectAltName DNS` field is managed.
- The certificate Subject DN will be set to whatever is specified in the PKCS#10 CSR.
- All limitations induced by the use of the Sectigo CMS REST Connector.

Create the PKI connector

1. Log in to Horizon Administration Interface.

2. Access PKI from the drawer or card: **PKI** > **PKI Connectors**.

3. Click on .

4. Select the correct PKI type.

5. Click on the next button

General tab

6. Fill in the common mandatory fields:

- **Connector Name*** *(string input)*:
  Choose a meaningful connector name allowing to identify the mapping between the PKI and the Certificate Profile. It must be unique and must not contain spaces.
- **Proxy** *(string select)*:
  If the PKI is not directly reachable from Horizon, you can set up an HTTP/HTTPS proxy to properly forward the traffic.
- **Queue PKI** *(string select)*:
  The PKI Queue used to manage the PKI Requests (enrollment, revocation).
- **Timeout** *(finite duration)*:
  Represents a predefined interval of time without a PKI response, when the time has passed "Horizon" will cease trying to establish the communication. Must be in valid finite duration.

7. Click on the next button

Details tab

8. Fill in all mandatory fields:

- **Customer URI*** *(string input)*:
  Enter the Customer URI. An integer is expected.
- **Organization ID*** *(int input)*:
  Enter the Organization ID.
- **Profile (Certificate Type)*** *(string input)*:
  Enter the Profile (Certificate Type). An integer is expected.
- **Retry interval** *(finite duration)*:
  Predefined interval of time before retrying to retrieve the certificate from Sectigo. Must be a valid finite duration. No default value is set.
- **Valid Days** *(finite duration)* :
  Certificate validity duration in days. Must be a valid finite duration. No default value is set.

9. Click on the next button.

Authentication tab

10. Fill in the PKI-authentication fields:

- **Login*** *(string input)*:
  Enter your Sectigo CMS login.
- **Password*** *(string input)*:
  Enter your Sectigo CMS password.

11. Click on the save button.

You can edit  , duplicate  or delete  the Sectigo CMS PKI connector.

5. Security

5.1. Local Accounts

This section details how to configure the EverTrust Horizon local accounts and set their password.

NOTE	Local accounts are useful to create technical accounts, such as required by `horizon-cli` for some scenarios (e.g. Scan/Discovery)

5.1.1. How to create local accounts

1. Log in to Horizon Administration Interface.

2. Access Local accounts from the drawer or card: **Security** > **Access Management** > **Local Accounts**.

3. Click on  .

4. Fill in the mandatory fields.

- **Identifier*** *(string input)*:
  Enter a meaningful identifier for the account holder. It will be used as a login to access to the solution.
- **Name** *(string input)*:
  Enter a meaningful name for the account holder.
- **Email** *(string input)*:
  Enter the account holder email.

5. Click on the save button.

5.1.2. How to set a password to a local account

1. Once a local account is created. Click on  .

2. Fill in the mandatory fields.

- **Password*** *(string input)*:
  Set a password.
- **Confirm password*** *(string input)*:
  Confirm the password.

3. Click on the save button.

You can edit  or delete  a local account. You can manage  a local account password.

NOTE	You can not delete yourself from local accounts.

5.2. Authorization

This section details how to configure the permissions granted to an account, either directly or through a configured role.

5.2.1. Prerequisites

According to the context, you might need to set up:

- Roles
- Local accounts

5.2.2. How to add an authorization manually or from a certificate

1. Log in to Horizon Administration Interface.

2. Access AUthoziations from the drawer or card: **Security** > **Access Management** > **Authorizations**.

3. Click on  .

4. Click on Add Authorization Manually

5. Fill the mandatory fields.

- Either:
  - Fill in an **Identifier*** *(string input or import)*:
    Enter a meaningful identifier. It can be either a local account identifier or an OpenID Connect identifier (usually email address).
  - Import a certificate by clicking on certificate button  .
- **Contact email** *(string input)*:
  Enter the contact email for the account.

6. Click on add button.

5.2.3. How to add an authorization from a search

1. Log in to Horizon Administration Interface.

2. Access AUthoziations from the drawer or card: **Security** > **Access Management** > **Authorizations**.

3. Click on  .

4. Click on Search and Add Authorization

5. Fill one of the fields.

- **Identifier*** *(string input)*:
  Enter the identifier of the account to look for.
- **Email*** *(string input)*:
  Enter the email of the account to look for.

6. Click on search button.

7. Choose the identifier you want to add.

8. Click on add button.

You can update  or delete  Authorization.

5.2.4. How to grant a permission

1. Click on  .

Role

2. Select a role previously created (if needed).

Team

3. Select a team previously created (if needed).

Configuration

You can build here a configuration permission. The permission follows the pattern: Section / Module / Right.

4. Click on add button.

5. Select a section, then a module, then a submodule if there is, and a right.

6. Click on add button (Don't forget to save).

7. Click on the save button if you are done.

Lifecycle

You can build here a lifecycle permission. The permission follows the pattern: Module / Profile / Right. You can further restrict the permission by adding a filter from the "Horizon Permission Query Language".

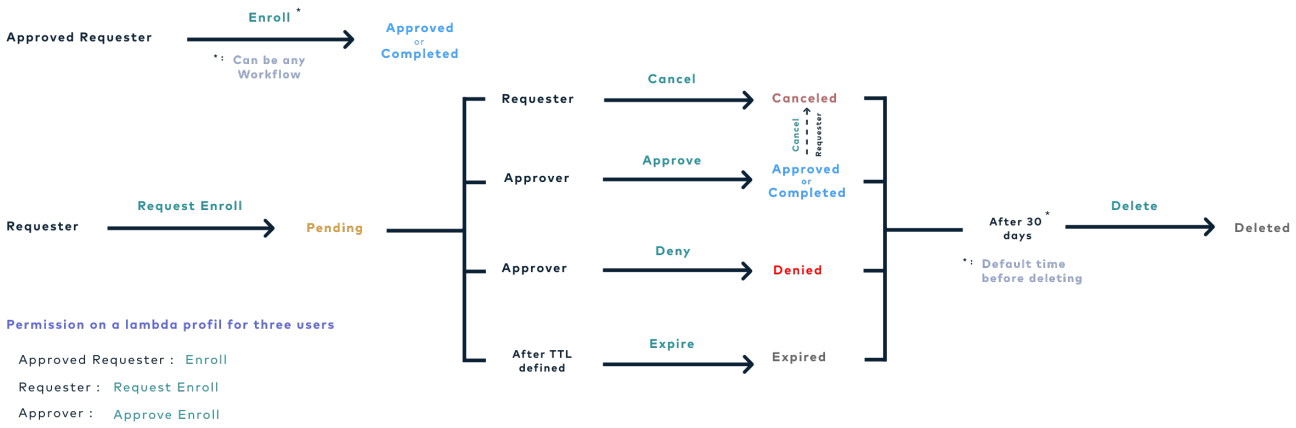
4. Click on add button.

5. Select a module, then a profile, and a right.

6. Click on add button. (don’t forget to save).

7. Click on the save button if you are done.

Horizon requests lifecycle:



Discovery

You can build here a discovery permission. The permission follows the pattern: Module / Discovery campaign name / Right.

4. Click on add button.

5. Select a module, then a campaign, and a right.

6. Click on add button. (don’t forget to save)

7. Click on the save button if you are done.

5.3. Roles

This section details how to configure the roles. Roles are groups of permissions that can be configured for authorizations.

5.3.1. How to create a role

1. Log in to Horizon Administration Interface.

2. Access Roles from the drawer or card: **Security** > **Access Management** > **Roles**.

3. Click on ⊕.

4. Fill in at least the mandatory fields.

- **Name*** *(string input)*:
  Enter a meaningful name.
- **Description** *(string input)*:

Enter a description.

6. configuration permissions

7. lifecycle permissions

8. discovery permissions

9. Click on the save button.

You can get the list of members  .

You can update  or delete  the Role.

5.4. Password Policies

This section details how to configure password policies that will be used by Horizon.

5.4.1. How to configure a Password Policy

1. Log in to Horizon Administration Interface.

2. Access Password Policies from the drawer or card: **Security** > **Password Policies**.

3. Click on  .

4. Fill in the mandatory fields.

- **Name***:
  Enter a meaningful password policy name;
- **Password range length*** *(int)*:
  Password length (0 is unlimited);
- **Minimum of lowercase** *(int)*:
  Minimum of lowercase characters in the password;
- **Minimum of uppercase** *(int)*:
  Minimum of uppercase characters in the password;
- **Minimum of digit** *(int)*:
  Minimum of digit in the password;
- **Minimum of special character** *(int)*:
  Minimum of special characters in the password;
- **Special characters accepted** *(string input)*:
  Whitelist of special characters accepted in the password.

5. Click on the save button.

You can update  or delete  the Password Policy.

CAUTION	You won't be able to delete a Password Policy if it is referenced in any other configuration element.

5.5. Identity Providers Configuration

This section details how to configure Identity Providers. Identity Providers are going to be used by Horizon to verify the identity of an end-user based on the authentication performed by an external authorization server.

5.5.1. How to configure an Identity Provider

1. Log in to Horizon Administration Interface.

2. Access Identity Providers from the drawer or card: **Security** > **Access Management** > **Identity Providers**.

3. Click on  .

General tab

4. Select an identity provider type. Currently only OpenID is supported

OpenID connect

5. Fill in all mandatory fields:

- **Name*** *(string input)*:
  Enter a meaningful identity provider name.
- **Provider metadata URL*** *(string input)*:
  Enter the OpenID Connect provider metadata URL.
- **Client ID*** *(string input)*:
  Identifier generated on the OpenID Connect IDP when setting up a new application (Horizon) to authenticate users on the identity provider.
- **Client Secret*** *(string input)*:
  Password associated to the aforementioned identifier (Client ID);
- **Scope*** *(string input)*:
  Scope used by Horizon during authentication on the identity provider to authorize access to user's details.
- Proxy *(string input)*:
  Proxy used to access Provider metadata URL, if any.
- **Timeout** *(finite duration)*:
  Timeout used for authentication on the identity provider. Must be a valid finite duration. By default 10 seconds.

- **Identifier Claim*** *(string input)*:
  Dynamic expression defining how to construct the identifier from the OpenID Connect claims. Claim names must be declared between `{{` and `}}` characters. For example, if the user identifier is contained in the `login` claim, then the configured value should be `{{login}}`.
- **Email Claim*** *(string input)*:
  Dynamic expression defining how to construct the user email from the OpenID Connect claims. Claim names must be declared between `{{` and `}}` characters. For example, if the user email is contained in the 'email' claim, then the configured value should be `{{email}}`. If the email is not available directly from the claims but can be computed from the 'login' claim by appending a domain, the configured value should be `{{login}}@evertrust.fr`.
- **Name Claim*** *(string input)*:
  Dynamic expression defining how to construct the username from the OpenID Connect claims. Claim names must be declared between `{{` and `}}` characters. For example, if the user name must be constructed as `family name, given name` and family name is available in the `family_name` claim, given name is available in the `given_name` claim, then the configured value should be `{{family_name}}, {{given_name}}`.
- **Enabled*** *(boolean)*:
  Enable/Disable the identity provider.
- **Enabled on `UI`*** *(boolean)*:
  Enable/Disable the identity provider on user interface.

Languages tab

* * **Language**: Please refer to Languages section to set up localized Identity Provider display name and description.

6. Click on the save button.

You can update  or delete  the Identity Provider.

CAUTION	You won't be able to delete an Identity Provider if it is referenced in any other configuration element.

5.6. Teams

This section details how to configure teams. Teams are groups of horizon objects owner (certificates, requests) and does not define permissions.

5.6.1. How to create a team

1. Log in to Horizon Administration Interface.

2. Access Teams from the drawer or card: **Security** > **Access Management** > **Teams**.

3. Click on  .

4. Fill at least the mandatory fields.

- **Name*** *(string input)*:
  Enter a meaningful name.
- **Description** *(string input)*:
  Enter a description.
- **Contact email** *(string input)*:
  Enter a valid mail.
- **Manager email** *(string input)*:
  Enter a valid mail.
- **Messaging tool** *(select)*:
  Select one of Webhook Messaging tools supported
- **URL** *(string input)*:
  Enter the webhook messaging URL for the team(used by Messaging notification)

5. Click on the save button.

You can get the list of members  .

You can update  or delete  the Team.

6. Notifications

This section explains how to configure notifications in Horizon. Horizon currently supports 2 types of notifications : mail notifications and instant messaging notifications.

6.1. Email Notifications

This section details how to configure the email notifications.

6.1.1. How to create a mail notification

1. Log in to Horizon Administration Interface.

2. Access emails from the drawer or card: **Notifications** > **Emails**.

3. Click on  .

4. Fill in all mandatory fields.

- **Name*** *(string input)*:
  Enter a meaningful mail notification name.
- **Event type*** *(select)*:
  Select the event type to notify (certificate or request).
- **Event*** *(select)*:
  Select the event to notify.
- **Retries in case of error** *(int)*:
  Select the number of times Horizon should retry to send the notification in case of error. The default value is set to 10.
- **From***: *(string input)*
  Enter the email address that will appear in the email "From" field.
- **To***: *(select multiple & input multiple)*
  Select one or several recipients. You may also enter an email address.
- **Subject*** *(string input)*:
  Enter the email subject. You may use dynamic attributes, that will be automatically replaced by the appropriate values upon email generation.
- **Body*** *(string input)*:
  Enter the email body. You may use dynamic attributes, that will be automatically replaced by the appropriate values upon email generation.
- **Is HTML** *(boolean)*:
  Sets whether the email body contains HTML code (true) or plain text (false). The default value is set to false.

NOTE	You can click on the "+" next to "How to use dynamic attributes" in order to get a range of possibilities from which one or more may be chosen.

*In case you selected any **Certificate** type event any **Request** type event but the **Enroll requests** ones:*

- **Attachments** *(list)*:
  Sets whether to attach the certificate to the email notification and which format to use for the attached certificate (if any).
  - Attach certificate (PEM) attaches the certificate under PEM format
  - Attach bundle (PEM) attaches the certificate as well as the entire trust chain used to sign it in PEM format
  - Attach certificate (PKCS#7) attaches the certificate under PKCS#7 format
  - Attach bundle (PKCS#7) attaches the certificate as well as the entire trust chain used to sign it in PKCS#7 format
  - Attach certificate (DER) attaches the certificate under DER format

*In case you selected *Certificate Expiration**:

- **Duration before certificate expiration causing the notification*** *(finite duration)*:
  Sets how long before certificate expiration the email notification should be sent. The default value is set to 5 days.
- **Run on renewed** *(boolean):*+ Sets whether the expiration notification should be sent even though the certificate has been renewed. Default value is set to false (if the certificate has been renewed, the notification will not be sent).

*In case you selected as an Event **Enroll request Approval** or **Recover request Approval**:*

- **Attach PKCS#12** (set at false) *(boolean)*:
  Sets whether the certificate in PKCS#12 format (certificate + private key encrypted by password) should be attached to the email. The default value is set to false.
- **Send email if** *(select unique)*:
  Select either Always - Centralized (Horizon generates the private key) - Decentralized (a CSR is provided to Horizon). The default value is set to Always.

*In case you selected as an Event **Enroll request Pending** or **Revoke request Pending** or **Recover request Pending** or **Update request Pending** or **Migrate Request Pending**:*

- **Duration after request submission causing the notification*** *(finite duration)*:
  Duration after request submission causing the notification to be sent, in case the request was not approved in the meantime. The default value is set to 5 days.

6. Click on the save button.

You can edit  , duplicate  or delete  the Email Notification .

6.2. Messaging

This section details how to configure the messaging notifications.

6.2.1. Prerequisites

You will need a webhook URL from the messaging tools in order to send notification.

6.2.2. How to create a Messaging notification

1. Log in to Horizon Administration Interface.

2. Access Messaging from the drawer or card: **Notifications** > **Messaging**.

3. Click on  .

4. Fill in all mandatory fields.

- **Name*** *(string input)*:
  Enter a meaningful email notification name.
- **Event type*** *(select)*:
  Select the event type to notify (certificate or request).
- **Event*** *(select)*:
  Select the event to notify.
- **Retries in case of error** *(int)*:
  Select the number of times Horizon should retry to send the notification in case of error.The default value is set to 10.
- **Timeout*** *(finite duration)*:
  The time before Horizon stop trying to connect to Webhook or Proxy.
- Proxy *(option)*:
  The HTTP/HTTPS proxy to use to reach the messaging tool, if any.
- **To*** *(select)*:
  Select one of:
  - Static
  - Teams webhook
- **Title*** *(string input)*:
  Enter the title of the instant messaging. You may use dynamic attributes, that will be automatically replaced by the appropriate values upon notification generation.
- **Body*** *(string input)*:
  Enter the body of the instant messaging. You may use dynamic attributes, that will be automatically replaced by the appropriate values upon notification generation.

NOTE	You can click on the "+" next to "How to use dynamic attributes" in order to get a

range of possibilities from which one or more may be chosen.

*In case you selected as an Event **Certificate Expiration**:*

- **Duration before certificate expiration causing the notification*** *(finite duration)*:
  Sets how long before certificate expiration the messaging notification should be sent. The default value is set to 5 days.

*In case you selected as an Event **Enroll request Pending** or **Revoke request Pending** or **Recover request Pending** or **Update request Pending** or **Migrate request Pending**:*

- **Duration after request submission causing the notification*** *(finite duration)*:
  Duration after request submission causing the messaging notification to be sent, in case the request was not approved in the meantime. The default value is set to 5 days.

6. Click on the save button.

You can edit  , duplicate  or delete  the Messaging Notification.

7. Discovery

This section details how to configure Discovery campaigns. An EverTrust Horizon Discovery campaign will contain all certificates discovered on a specific scope.

CAUTION

A discovered certificate can be:

- An unknown certificate.
  > All certificate information will be stored and this certificate will appear as an ' `unmanaged` ' certificate.
- An already discovered certificate (due to another Discovery campaign).
  > Discovery campaign metadata will be added to the existing certificate.
- A managed certificate.
  > Discovery campaign metadata will be added to the existing certificate.

7.1. How to create a Discovery Campaign

1. Log in to Horizon Administration Interface.

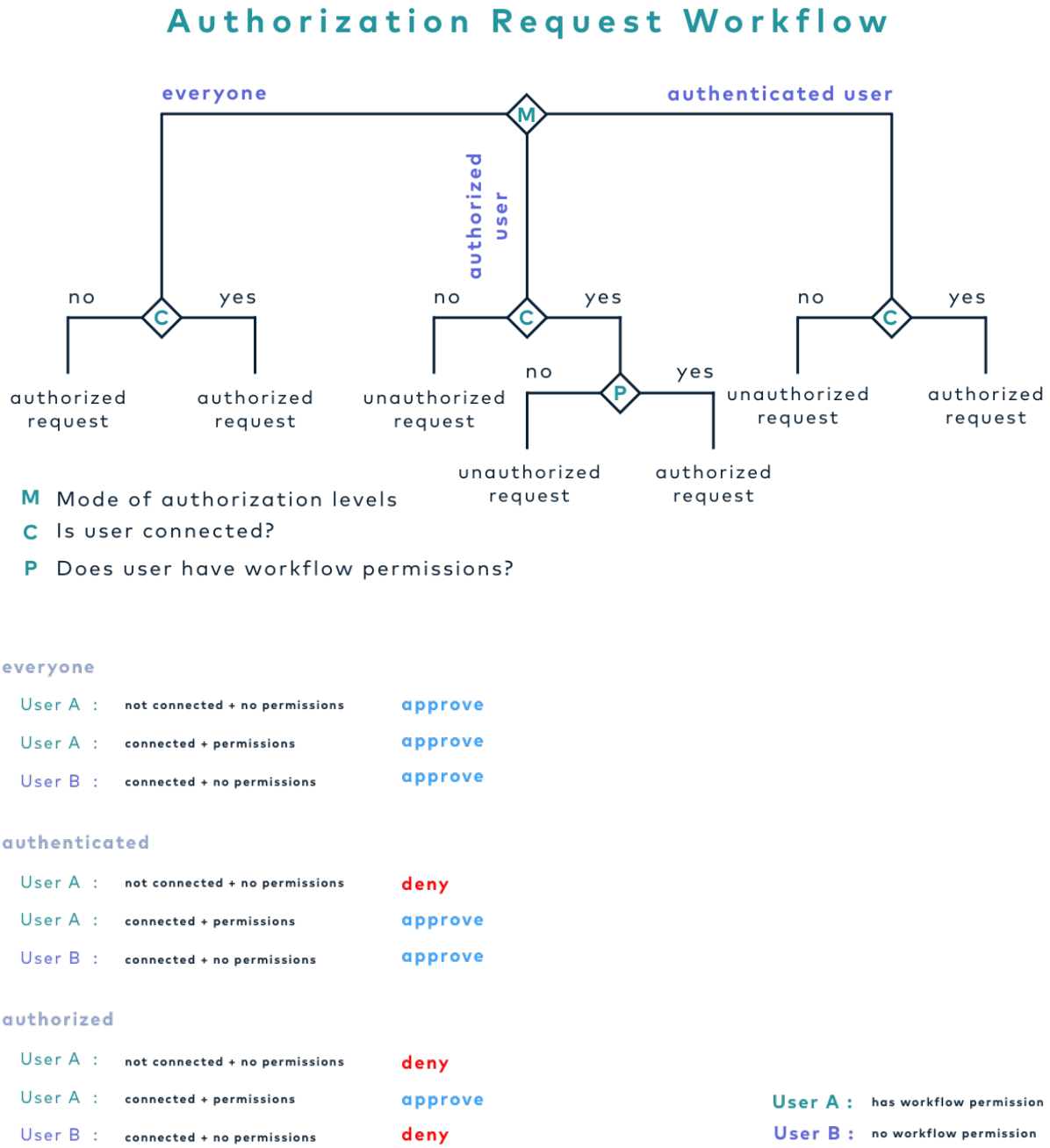
2. Access Discovery from the drawer or card: **Discovery**.

3. Click on  .

4. Fill in all mandatory fields.

General tab

- **Campaign name*** *(string input)*:
  Enter a meaningful Discovery campaign name.
- **Description** *(string input)*:
  Enter Discovery campaign description.
- **Enabled** *(boolean)*:
  Enable/Disable this Discovery campaign.
- **Search** *(select)*:
  Select an authorization level to search this Discovery campaign.
- **Feed** *(select)*:
  Select an authorization level to feed this Discovery campaign.



- **Log event on success*** *(boolean)*:
  Enable/Disable discovery event on success.
- **Log event on failure*** *(boolean)*:
  Enable/Disable discovery event on failure.
- **Log event on warning*** *(boolean)*:
  Enable/Disable discovery event on warning.

Host tab

- **Hosts** *(string input or int)*:
  Specify the target to scan. Can be hostname(s), IP address(es), IP range or CIDR address(es). It is possible to add several hostnames separated by commas.

Port tab

- **Ports** *(string input or int)*:
  Enter the port(s) to scan on hosts. It is possible to add several ports separated by commas or to add a port range separated by an hyphen (ex : 1-1000 to go from 1 to 1000).

NOTE	Hosts and ports should only be set if you intend to perform a network scan using `horizon-cli` in order to discover the certificates. These parameters are ignored in all other discovery modes (local scan, third party import).

6. Click on the save button.

You can edit  , flush  or delete  the Discovery.

7.2. How to flush a Discovery Campaign

Flushing a Discovery campaign is the action to remove Discovery campaign reference from all discovered certificates.

CAUTION	There are three different cases: • If the certificate is not managed by Horizon (only discovered by a Discovery campaign) AND only referenced by the campaign you are willing to flush → The certificate will be removed from the Horizon database. • If the certificate is not managed by Horizon but is referenced by at least another Discovery campaign → The certificate will NOT be removed from the database and only the Discovery metadata will be removed from the certificate. • If the certificate is managed by Horizon → Only the Discovery metadata will be removed from the certificate.

1. Log in to Horizon Administration Interface.

2. Access Discovery from the drawer or card: **Discovery**.

3. Click on  .

4. Click on the Confirm button to perform the flush.

8. Protocols

8.1. ACME

8.1.1. Introduction

This section details how to configure and consume the ACME protocol.

Horizon implements an ACME service respecting the RFC 8555 and more specifically the following lifecycle workflows:

- Enrollment;
- Renewal (which is equivalent to an enrollment);
- Revocation.

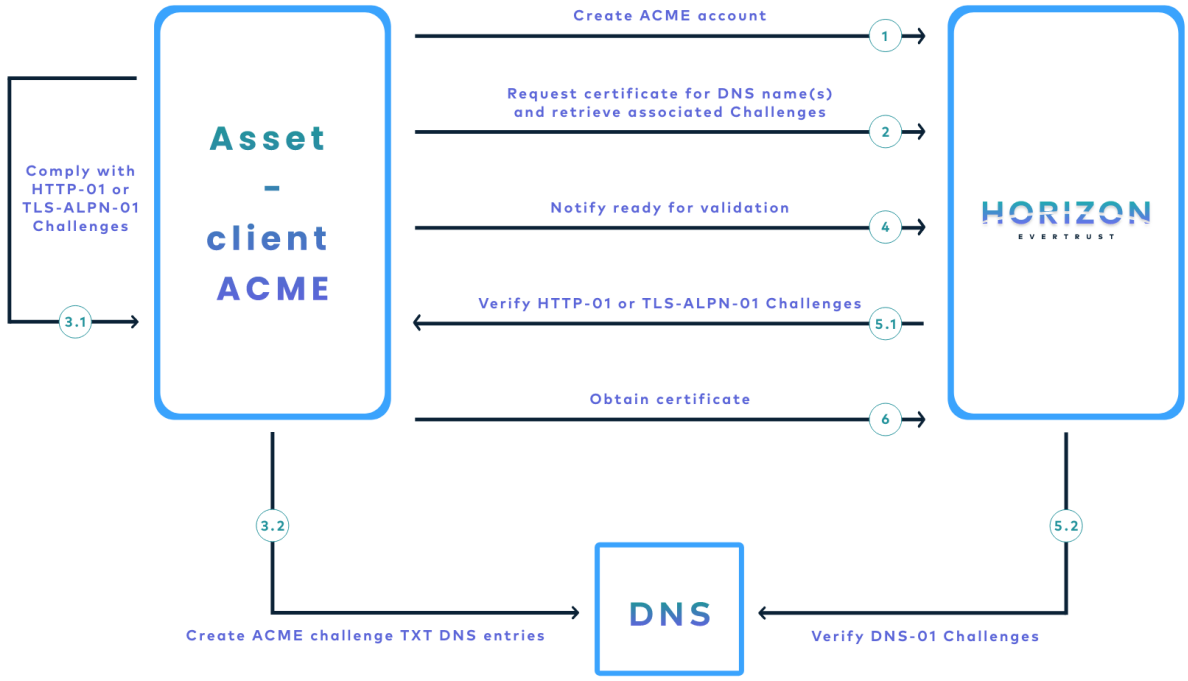
Managing certificate lifecycle through the ACME protocol involves up to three components:

- Horizon as the ACME endpoint;
- An asset executing an ACME client or directly integrating the ACME protocol;
- When ACME validation is ' `dns-01` ', DNS server(s).

NOTE	ACME validation modes will be detailed later on.

The protocol paradigm can be described as follows: '**if the asset can prove it has authority on the DNS names (called identifiers in ACME) it is requesting for, the certificate should be automatically enrolled / renewed**', which is basically equivalent to a **Domain Validation**.

The following schema is a simplified workflow of an ACME enrollment:



The protocol is based on the notion of challenge and offers three validation modes to actually verify challenges and prove that the asset owns authority on the requested DNS name(s), i.e. ACME identifiers:

- `http-01`: For each requested identifier, Horizon will validate the challenge by connecting back in **HTTP** on the configured `http-01` *validation port* (TCP/80 by default) and retrieve the response to the challenge;
- `tls-alpn-01`: For each requested identifier, Horizon will validate the challenge by connecting back in **HTTPS** on the configured `tls-alpn-01` *validation port* (TCP/443 by default) and extract the response to the challenge from an **ALPN** extension in the asset / client **HTTPS** response;
- `dns-01`: For each requested identifier, Horizon will validate the challenge through a DNS request and look for a specific TXT entry containing the response corresponding to the challenge for the considered identifier.

Therefore, validation modes have the following constraints:

- `http-01 and tls-alpn-01`:
  - Horizon must be able to access the asset on the validation port;
  - The validation port must be available and opened on the asset;
- `dns-01`: the ACME client must be configured with DNS credentials owning the permission to create TXT records on the requested domain(s).

NOTE

For `http-01` and `tls_alpn-01` validation modes, it is possible to configure an HTTP proxy to proxify the ACME validation tentative(s). Using an HTTP proxy is useful when `http-01` and/or `tls-alpn-01` validation need to be performed on asset(s) hosted within a DMZ where incoming network streams must be limited. In this scenario, an HTTP proxy is configured to relay ACME validations coming from the Horizon

nodes within the DMZ and a unique incoming stream needs to be open to allow communication from Horizon node to the HTTP proxy.

The choice of the validation mode to use mainly depends on the architecture. Here are the EverTrust recommendations:

- If the requester is not the asset, prefer the `dns-01` validation mode;
- If the requester is the asset:
  - If the asset is reachable from Horizon nodes, prefer the `http-01`;
  - If the asset is not reachable from Horizon nodes, prefer the `dns-01`;
- `tls-alpn-01` is the most complicated validation mode to implement and therefore should only be used when no other validation mode is an option.

8.1.2. Qualified ACME clients

EverTrust qualifies the following ACME clients for any release of the Horizon product:

- Linux ACME clients:
  - acme.sh
  - certbot
  - lego
- Windows ACME client:
  - lego
  - WinCertes: this open source client is developed and maintained by EverTrust, therefore officially supported
- Kubernetes: cert-manager

NOTE If an ACME client is not listed above, it does not necessarily mean that the client will not work with Horizon, only that the client is not included in the list of clients tested in Horizon's continuous integration test cases.

8.1.3. ACME Profile

This section details how to configure an ACME Profile.

Prerequisites

On Horizon side, you need to set up:

- PKI Connector

How to configure ACME Profile

1. Log in to Horizon Administration Interface.

2. Access ACME Profile from the drawer or card: **Protocols** > **ACME**.

3. Click on  .

4. Fill in the mandatory fields.

General

- **Name*** *(string input)*:
  Enter a meaningful connector name. It must be unique for each profile. Horizon uses the name to identify the profile. As the name will be part of a URL, it is advisable to use only lower case letters and dashes.
- **Enabled*** *(boolean)*:
  Indicates whether the profile is enabled or not. The default value is set to true.
- **PKI Connector** *(select)*:
  Select a PKI connector previously created.
- **Max certificate per holder** *(int)*:
  If specified, defines the maximum number of active certificates for a given Holder. If the number of active certificates exceeds this parameter, then the oldest certificate(s) above the limit will be automatically revoked.

Validations

- **Validation Methods** *(select)*:
  Select the authorized ACME validation method(s) on the considered profile (`HTTP-01` and/or `TLS-ALPN-01` and/or `DNS-01`).
- `HTTP_01` **validation port** *(int)*:
  HTTP port to perform the `http-01` validation. The default value is set to 80.
- `TLS-ALPN_01` **validation port** *(int)*:
  HTTPS port to perform the `tls-alpn-01` validation. The default value is set to 443.
- **Challenge verification attempts*** *(int)*:
  Specify the number of time Horizon should try to validate an ACME challenge. Th default value is set to 3.
- **Challenge verification retry delay*** *(finite duration)*:
  Specify the time duration Horizon should wait between two consecutive validations for the same challenge. The default value is set to 3 seconds.
- **Proxy** *(select)*:
  Specify an HTTP proxy to use when performing `http-01` or `tls-alpn-01` validations.
- **Timeout*** *(finite duration)*:
  Specify the time duration Horizon should wait when performing `http-01`, `tls-alpn-01` or `dns-01` validations.

Requests management

- **Authorized short name** *(boolean)*:
  Specify if using short name is authorized when requesting certificate. If set to yes, one verifiable FQDN must be requested for each specified short name. The default value is set to false.
- **Authorized empty contact** *(boolean)*:
  Specify if an ACME account can be registered without specifying a contact email address. Default to false.
- **Default contacts mail** *(string input multiple)*:
  Specify a list of default contact email addresses when registering an ACME account with no specified contact email address.
- **Max DNS name** *(int)*:
  If specified, enforce the maximum number of requested DNS name(s).

Meta

- **Is required terms of service** *(boolean)*:
  Specify if explicitly agreeing to the terms of service is required when registering an ACME account. The default value is set to false.
- **Terms of service** *(string input)*:
  Specify an URL identifying the current terms of service.
- **Website** *(string input)*:
  Specify an HTTP or HTTPS URL locating a website providing more information about the ACME server.
- **CAA Identities** *(string input)*:
  The hostnames that the ACME server recognizes as referring to itself for the purposes of CAA record validation as defined in RFC6844.

Self Permissions

- **Revoke** *(boolean)*:
  Specify whether the certificate's owner is authorized to revoke the certificate with no validation workflow. Set to false by default.
- **Request Revoke** *(boolean)*:
  Specify whether the certificate's owner is authorized to request the revocation of the certificate. Set to false by default.
- **Update** *(boolean)*: Specify whether the certificate's owner is authorized to update the certificate with no validation workflow. The default value is set to false.
- **Request Update** *(boolean)*:
  Specify whether the certificate's owner is authorized to request certificate's update. The default value is set to false.

You can further configure the profile using the Common configuration for profile and Notification tabs.

5. Click on the save button.

You can edit , duplicate or delete the ACME Profile.

CAUTION	You won't be able to delete an ACME Profile if it is referenced somewhere else.

8.1.4. ACME client usages

This section details how to use the most common Linux and Windows ACME clients.

Linux ACME clients

This section details how to use the **acme.sh** and **certbot** ACME clients.

Overview

Certbot is able to run on any recent UNIX-like operating system equipped with Python 2.7 or 3.4+, while acme.sh can also run on any recent Linux distribution running either bash, dash or sh.

They both fully support the latest ACMEv2 protocol including its main latest feature: wildcard certificates (*.example.com).

Both clients supports different modes for obtaining a certificate and in some cases automatically installing it.

The following tables lists the different modes for each clients:

Modes	certbot	acme.sh	Notes
apache	Y	Y	Obtains and automatically installs a certificate using the running Apache server. (For acme.sh, this mode will only obtain a certificate without installing it)
nginx	Y	Y	Obtains and automatically installs a certificate using the running NGINX server. (For acme.sh, this mode will only obtain a certificate without installing it)
webroot	Y	Y	Obtains a certificate by writing to the webroot directory of an already running web server
standalone	Y	Y	Uses a "standalone" web server managed by Certbot or acme.sh. This mode is useful on system with no web servers or if using the running web server is not desired
DNS	Y	Y	This mode automates obtaining a certificate by modifying a DNS record to prove the control over a domain
`tls-alpn`	N	Y	Uses a TLS server to validate the control over a domain

Requesting a certificate

Both clients must be started using administrative privileges (`sudo`), except for acme.sh when using the webroot or DNS modes.

Each client requires only a few parameters to request a certificate.

acme.sh parameters:

Parameter	Description
`-issue`	Obtain or renew a certificate, but does not install it
`-w [VALUE]`	Path of the server's webroot folder
`-d [VALUE]`	The domain(s) to enroll.

certbot parameters:

Parameter	Description
`certonly`	Obtain or renew a certificate, but does not install it
`webroot`	Place files in a server's webroot folder for authentication
`-w [VALUE]`	Path of the server's webroot folder
`-d [VALUE]`	The domain(s) to enroll.

Requesting a certificate for Apache using certbot:

```
(sudo) certbot run --apache --no-eff-email --agree-tos --server <Horizon ACME
endpoint, example: https://horizon.evertrust.fr/acme/profile1/directory> -m <contact
email address, example: kma@evertrust.fr> --domain <DNS name, example:
apache.evertrust.fr>
```

Where:

- `--apache`: Enables the Apache mode
- `--no-eff-email`: Does not share your email address with EFF
- `--agree-tos`: Explicitly agrees to the terms of service
- `--server`: Horizon ACME profile endpoint
- `-m`: Contact email address
- `--domain`: Requested DNS name (can be specified several times)

Requesting a certificate for nginx using certbot:

```
(sudo) certbot run --nginx --no-eff-email --agree-tos --server <Horizon ACME endpoint,
example: https://horizon.evertrust.fr/acme/profile1/directory> -m <contact email
address, example: kma@evertrust.fr> --domain <DNS name, example: nginx.evertrust.fr>
```

Where:

- `--nginx`: Enables the nginx mode
- `--no-eff-email`: Does not share your email address with EFF
- `--agree-tos`: Explicitly agrees to the terms of service
- `--server`: Horizon ACME profile endpoint
- `-m`: Contact email address
- `--domain`: Requested DNS name (can be specified several times)

Requesting a certificate for nginx using acme.sh:

```
(sudo) acme.sh --issue --nginx --server <Horizon ACME endpoint, example:
https://horizon.evertrust.fr/acme/profile1/directory> --accountemail <contact email
address, example: kma@evertrust.fr> -d <DNS name, example: nginx.evertrust.fr>
```

Where:

- `--issue`: Specifies that this is a certificate request
- `--nginx`: Enables the nginx mode
- `--server`: Horizon ACME profile endpoint
- `--accountemail`: Contact email address
- `-d`: Requested DNS name (can be specified several times)

Requesting a certificate in standalone mode using certbot:

```
(sudo) certbot certonly --standalone --no-eff-email --agree-tos --server <Horizon ACME
endpoint, example: https://horizon.evertrust.fr/acme/profile1/directory> -m <contact
email address, example: kma@evertrust.fr> --domain <DNS name, example:
apache.evertrust.fr>
```

Where:

- `--standalone`: Enables the standalone mode, i.e. certbot will start a local web server to server the response
- `--no-eff-email`: Does not share your email address with EFF
- `--agree-tos`: Explicitly agrees to the terms of service
- `--server`: Horizon ACME profile endpoint

- `-m`: Contact email address
- `--domain`: Requested DNS name (can be specified several times)

Requesting a certificate in standalone mode using acme.sh:

```
(sudo) acme.sh --issue --standalone --server <Horizon ACME endpoint, example:
https://horizon.evertrust.fr/acme/profile1/directory> --accountemail <contact email
address, example: kma@evertrust.fr> -d <DNS name, example: apache.evertrust.fr>
```

Where:

- `--issue`: Specifies that this is a certificate request
- `--standalone`: Enables the standalone mode, i.e. acme.sh will start a local web server to server the response
- `--server`: Horizon ACME profile endpoint
- `--accountemail`: Contact email address
- `-d`: Requested DNS name (can be specified several times)

Revoking a certificate

Revoking a certificate using certbot:

```
(sudo) certbot revoke --cert-path <path of the certificate to revoke> --server
<Horizon ACME endpoint, example: https://horizon.evertrust.fr/acme/profile1/directory>
```

Where:

- `--cert-path`: Specifies the path of the certificate to revoke
- `--server`: Horizon ACME profile endpoint

Revoking a certificate using acme.sh:

```
(sudo) acme.sh --server <Horizon ACME endpoint, example:
https://horizon.evertrust.fr/acme/profile1/directory> --revoke -d <DNS name, example:
apache.evertrust.fr>
```

Where:

- `--server`: Horizon ACME profile endpoint
- `-d`: DNS name of the certificate to revoke

8.1.5. Windows ACME clients

This section details how to use the **WinCertes** ACME client.

Overview

WinCertes is a simple and efficient CLI-based client made to run on any Windows Server (> Windows Server 2008 R2 SP1 (64 bits)) and running .NET 4.6.1 or higher.

The client fully supports ACMEv2 including its latest feature, along with the support of wildcard certificates (*.example.com).

WinCertes eases certificate installation and renewal by automatically binding them to the appropriate web site on IIS and by creating a Scheduled Task that will check the expiration date of the certificates and trigger a renewal if necessary.

WinCertes offers the possibility to launch a PowerShell script upon the successful retrieval of a certificate. This feature enables advanced deployment on Exchange or multi-servers for instance.

The client supports two validation modes for validating the identity of the certificate requester:

1. HTTP challenge validation
    - With the ability to support the running IIS web server or to use an embedded standalone web server for easier configuration.
2. DNS challenge validation
    - Support for Windows DNS Server
    - Support for `acme-dns`

Requesting a certificate

To request a certificate using WinCertes, the Windows command line (`cmd.exe`) must be run as Administrator.

Then WinCertes requires only a few parameters to request a certificate:

Parameter	Description
`-d [VALUE]`	The domain(s) to enroll
`-w`	toggle the local web server use and sets its ROOT directory (default `c:\inetpub\wwwroot`). Activates HTTP validation mode.
`-b [VALUE]`	The name of the IIS web site to bind the certificate to
`-p`	Used to make WinCertes create a Scheduled Task to handle certificate renewal

There are many more options to customize the requests to specific needs.

Requesting a certificate for IIS using WinCertes:

```
(as administrator) wincertes -s <Horizon ACME endpoint, example:
https://horizon.evertrust.fr/acme/profile1/directory> -w -b <IIS Site Name, example:
```

```
"Default Web Site"> -p -e <contact email address, example: kma@evertrust.fr> -d <DNS
name, example: iis.evertrust.fr>
```

Where:

- `-s`: Horizon ACME profile endpoint
- `-w`: Enables standalone mode, i.e. WinCertes will start a local web server to serve the response
- `-b`: IIS Web Site name
- `-p`: Registers a scheduled task to enable certificate automated renewal
- `-e`: Contact email address

8.2. EST

This section refers to the EST protocol, as described by RFC 7030.

8.2.1. EST Profile

This section details how to configure the EST Profile

8.2.2. Prerequisites

PKI

How to configure EST Profile

1. Log in to Horizon Administration Interface.

2. Access EST Profile from the drawer or card: **Protocol** > **EST**.

3. Click on  .

4. Fill in the mandatory fields.

General

- **Name*** *(string input)*:
  Enter a meaningful profile name. It must be unique for each profile. Horizon use the name to identify the profile.
- **Enabled** *(boolean)*:
  Tells whether the profile is enabled or not. The default value is set to true.
- **PKI** *(select)*:
  Select a PKI Connector previously created.
- **Max certificates per holder** *(int)*:
  When specified, define the maximum number of active certificates for a given Holder.

Crypto Policy

- **Decentralized enrollment** *(boolean)*:
  Tells whether the profile should be used with a decentralized enrollment mode, i.e CSR (PKCS#10) signing by the PKI. The default value is set to true.
- **Centralized enrollment** *(boolean)*:
  Tells whether the profile should be used with a centralized enrollment, i.e providing a PKCS#12. The default value is set to false.
  - **Private key escrowing** *(boolean)*:
    Tells whether the private key should be escrowed by Horizon. (only for **Centralized enrollment**) The default value is set to false.
  - **Key type** *(select)*:
    Select the type of key to generate when using centralized enrollment mode.
  - **Password policy for PKCS#12 password*** *(select)*:
    Select a password policy previously created.
  - **Store encryption type** *(select)*:
    Select from the list the encryption type. The default value is set to DES_AVERAGE.
  - **Show PKCS#12 Password On Recover** *(boolean)*:
    Tells whether the PKCS#12 password should be displayed on recover. Enabled when Private key escrowing is set to on. The default value is set to false.
  - **Show PKCS#12 On Recover** *(boolean)*:
    Tells whether the PKCS#12 should be displayed on recover. Enabled when Private key escrowing is set to on. The default value is set to false.

Authorization and validation

- **Authorization mode** *(select)*:
  Select from the list.
- **Authorized**:
  - **Enabled whitelist** *(boolean)*:
    Tells whether whitelist is enabled or not. The default value is set to false.
  - **CA*** *(select)*:
    Select a Certificate Authority previously created.
- **X509**:
  - **Enrollment CAs** *(select)*:
    Available only if mode at x509. Select a Certificate Authority previously created.
  - **Enabled whitelist** *(boolean)*:
    Tells whether whitelist is enabled or not. The default value is set to false.
  - **CA*** *(select)*:
    Select a Certificate Authority previously created.
- **Challenge**:
  - **Password policy** *(select)*:

Select a password policy previously created. It is used for the challenge generation.

- **Enabled whitelist** *(boolean)*:
  Tells whether whitelist is enabled or not. The default value is set to false.
- **CA*** *(select)*:
  Select a Certificate Authority previously created.

Renewal management

- **Renewal period:** *(finite duration)*:
  Must be a valid finite duration.
- **Renewal CAs** *(select)*:
  Select a Certificate Authority previously created.
- **Revoke on renew** *(boolean)*:
  The previous certificate will be revoked on renew if true. The default value is set to false.
- **Revocation reason** *(select)*:
  Select the reason from the list. Available only if "revoke on renew" value is set to true.

Self Permissions

- **Revoke** *(boolean)*:
  Tells whether self revoke permission is granted or not. The default value is set to false.
- **Request Revoke** *(boolean)*:
  Tells whether self request revoke permission is granted or not. The default value is set to false.
- **Update** *(boolean)*:
  Tells whether self update permission is granted or not. The default value is set to false.
- **Request Update** *(boolean)*:
  Tells whether self request update permission is granted or not. The default value is set to false.
- **Recover** *(boolean)*:
  Tells whether self recover permission is granted or not. The default value is set to false.
- **Request recover** *(boolean)*:
  Tells whether self request recover permission is granted or not. The default value is set to false.

You can further configure the profile using the Common configuration profile and Notification tabs.

5. Click on the save button.

You can edit  , duplicate  or delete  the EST Profile.

CAUTION	You won't be able to delete a EST Profile if this one is referenced somewhere else.

8.3. SCEP

This section refers to the SCEP protocol, as described by RFC 8894.

8.3.1. SCEP Profile

This section details how to configure the SCEP Profile

8.3.2. Prerequisites

PKI Connector	SCEP Authority

How to configure SCEP Profile

1. Log in to Horizon Administration Interface.

2. Access SCEP Profile from the drawer or card: **Protocol** > **SCEP**.

3. Click on  .

4. Fill in the mandatory fields.

General

- **Name*** *(string input)*:
  Enter a meaningful profile name. It must be unique for each profile. Horizon use the name to identify the profile.
- **Enabled** *(boolean)*:
  Tells whether the profile is enabled or not. The default value is set to true.
- **PKI Connector*** *(select)*:
  Select a PKI connector previously created.
- **Max certificate per holder** *(int)*:
  When specified, define the maximum number of active certificates for a given Holder.
- **Enabled NDES emulation mode** *(boolean)*:
  Tells whether the NDES emulation mode is enabled or not. The defaults value is set to false.
- **DN Whitelist*** *(boolean)*:
  Tells whether the DN whitelist is enabled or not. The default value is set to false.

SCEP protocol parameters

- **Mode*** *(select)*:
  Choose from the two modes RA or CA. The default value is set to RA.
- **SCEP Authority*** *(select)*:
  Select a previously created SCEP Authority.
- **CAPS*** *(select)*:
  Select a caps from the list. The default value is set to SHA.
- **Encryption algorithm*** *(select)*:
  Select an encryption algorithm from the list.

- **Password policy** *(select)*:
  Select a previously created password policy. It is used for the challenge generation.

Renewal management

- **Renewal period** *(finite duration)*:
  Must be in valid finite duration.
- **Revocation on SCEP renew?*** *(boolean)*:
  The previous certificate will be revoked on renew if true. The default value is set to false.
- **Revocation reason*** *(select)*:
  Select the reason from the list. Available only if "revocation on SCEP renew" value is set to true.

Self Permissions

- **Revoke** *(boolean)*:
  Tells whether self revoke permission is granted or not. The default value is set to false.
- **Request Revoke** *(boolean)*:
  Tells whether self request revoke permission is granted or not. The default value is set to false.
- **Update** *(boolean)*:
  Tells whether self update permission is granted or not. The default value is set to false.
- **Request Update** *(boolean)*:
  Tells whether self request update permission is granted or not. The default value is set to false.

You can further configure the profile using the Common configuration profile and Notification tabs.

5. Click on the save button.

You can edit , duplicate or delete the SCEP Profile.

CAUTION	You won't be able to delete a SCEP Profile if this one is referenced somewhere else.

8.4. WCCE

8.4.1. Introduction

This section details how to configure and consume the Windows Client Certificate Enrollment (WCCE) protocol.

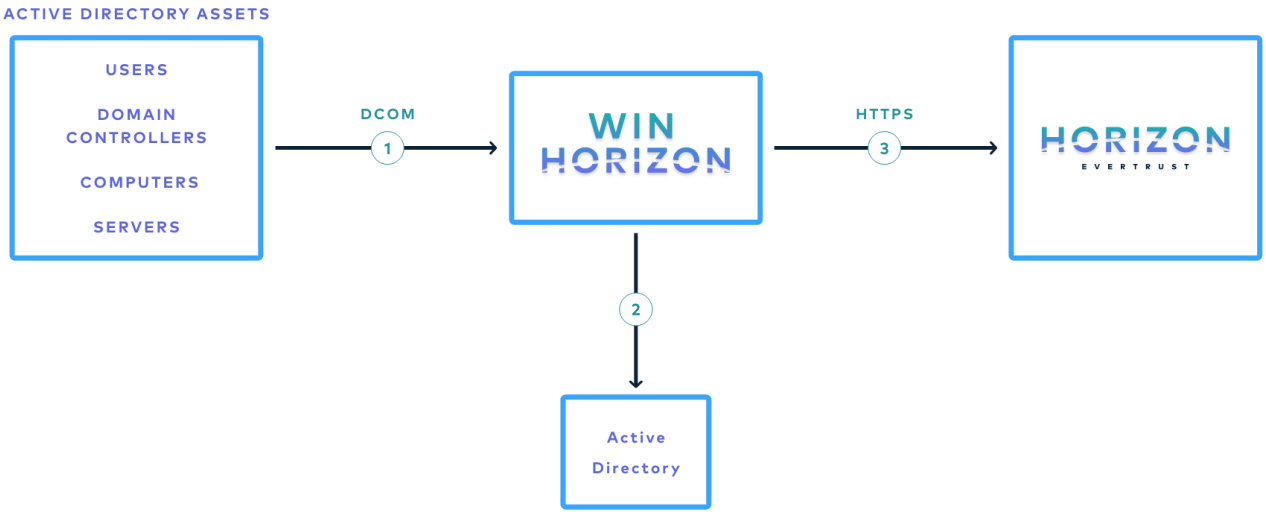
Managing certificate lifecycle through the WCCE protocol involves up to three components:

- Active Directory asset (domain controller, server, workstation, user) as WCCE Client;
- WinHorizon as the Active Directory enrollment service;
- Horizon as the WCCE proxy;

NOTE WCCE enrollment modes will be detailed later on.

The protocol paradigm can be described as follows: '**every Windows Active Directory member (machines, users) can use DCOM interfaces to interact with a CA to request certificate enrollment**'.

The following schema is a simplified workflow of an WCCE enrollment:



The protocol is based on the notion of Active Directory membership and configuration. Active Directory clients (such as machines and users) having rights on **Microsoft Certificate Templates** can use Active Directory **enrollment service** through DCOM interface to request certificate enrollment.

Horizon supports different WCCE enrollment modes:

- **Entity**: Certificate's elements are built using Active Directory content;
- **Enrollment On Behalf of Others (EOBO)**: Certificate signing request (CSR) is signed by one/many Certificate Enrollment Agent(s);
- **Trust request**: Certificate signature request (CSR) content is fully trust and certificate will be created using its content.

NOTE For **Enrollment On Behalf of Others (EOBO)** enrollment mode, it is possible to configure a whitelist of **Authorized CAs** trusted as issuers of enrollment agent certificates.

8.4.2. Windows official resources

EverTrust WCCE implementation is based on official WCCE documentation provided by Microsoft:

- MS-WCCE: Windows Client Certificate Enrollment Protocol

8.4.3. Prerequisites

- WinHorizon should be installed using WinHorizon installation guide;
- WinHorizon and Active Directory should be configured using WinHorizon administration guide.

8.4.4. WCCE Forest

The first step is to register WCCE Forest on which you want to use WCCE protocol through Horizon.

Uses

- MSAD Connector

How to configure WCCE Forest

1. Log in to Horizon Administration Interface.

2. Access WCCE Forest from the drawer or card: **Protocol** > **WCCE** > **Forest**.

3. Click on  .

4. Fill the mandatory fields.

- **Forest Name*** *(string input)*: Enter the Active Directory forest name.

5. Click on the save button.

You can duplicate  or delete  the WCCE Forest.

CAUTION	You won't be able to delete an WCCE Forest if it is referenced somewhere else.

8.4.5. WCCE Profile

*The second step details how to create and configure a WCCE Horizon profile. This profile is an **internal** Horizon profile.*

Uses

WCCE Template Mapping	WCCE Scheduled Tasks

Prerequisites

PKI

How to configure a WCCE Profile

1. Log in to Horizon Administration Interface.

2. Access WCCE Profile from the drawer or card: **Protocol** > **WCCE** > **Profiles**.

3. Click on  .

4. Fill the mandatory fields.

General

- **Name*** *(string input)*:
  Enter a meaningful profile name. It must be unique for each profile. Horizon uses the name to identify the profile. As the name will be part of an URL, it is advisable to use only lower case letters and dashes.
- **Enabled*** *(boolean)*:
  Indicates whether the profile is enabled or not. The default value is set to true.
- **Max certificate per holder** *(int)*:
  If specified, defines the maximum number of active certificates for a given Holder. If the number of active certificates exceeds this parameter, then the oldest certificate(s) above the limit will be automatically revoked.
- **PKI** *(select)*:
  Select a PKI connector previously created.

Self Permissions

- **Revoke** *(boolean)*:
  Specify whether the certificate's owner is authorized to revoke the certificate with no validation workflow. The default value is set to false.
- **Request Revoke** *(boolean)*:
  Specify whether the certificate's owner is authorized to request the revocation of the certificate. The default value is set to false.
- **Update** *(boolean)*: Specify whether the certificate's owner is authorized to update the certificate with no validation workflow. The default value is set to false.
- **Request Update** *(boolean)*:
  Specify whether the certificate's owner is authorized to request certificate's update. The default value is set to false.
- **Recover** *(boolean)*:
  Specify whether the certificate's owner is authorized to recover the certificate with no validation workflow. The default value is set to false.
- **Request recover** *(boolean)*:
  Specify whether the certificate's owner is authorized to request certificate's recover. The default value is set to false.

You can further configure the profile using the Common configuration profile and Notification tabs.

5. Click on the save button.

You can edit  , duplicate  or delete  the WCCE Profile .

CAUTION	You won't be able to delete a WCCE Profile if this one is referenced somewhere else.

8.4.6. WCCE Template Mapping

The third and last step is to configure mapping between Microsoft Certificate Template configured on Active Directory and Horizon WCCE profile. A mapping is created using a specific enrollment mode. As a result of this mapping, every Microsoft Certificate Template can issue certificate from different PKI (using PKI connector of WCCE profile associated to Microsoft Certificate Template).

Prerequisites

WCCE Forest	WCCE Profile

How to configure WCCE Template Mapping

1. Log in to Horizon Administration Interface.

2. Access WCCE Forest from the drawer or card: **Protocol** > **WCCE** > **Forest**.

3. Identify the section corresponds to the forest for which you want to add mapping. Click on + button.

4. Fill the mandatory fields.

- **Microsoft Template Name*** *(string input)*:
  Enter the Microsoft Certificate Template name created on Active Directory side.
- **Enrollment mode** *(select)*:
  Specify the enrollment mode of this mapping.
- **EOBO CAs** *(select)*:
  Specify the CA(s) to use for EOBO enrolment.
- **Profile*** *(select)*:
  Select a previously created WCCE profile.

5. Click on the save button.

You can edit  or delete the WCCE Template mapping.

8.4.7. WCCE test enrollment

This section details how to use the Microsoft Management Console (MMC) to manually retrieve a certificate through WCCE using different enrollment modes. If you want to enroll **machine certificate** you need to perform the following actions using Administrator Account.

1. Launch `mmc.exe`

2. Click on **File** > **Add/Remove or Remove Snap-ins**

3. On the left panel, click on **Certificates** then **Add**

NOTE	If you don't have administrative privileges, the **User certificate store** will be automatically chosen. If your account has administrative privileges, it will be prompted a window to choose Microsoft Certificate Store to use. If you want to enroll **User** certificate please chose **My user account**. If you want to enroll **Machine** certificate (computer or IIS for example) please chose **Computer account**.

4. Navigate to **Personal** > **Certificates**

5. Right click on Windows and chose **All tasks** > **Request certificate**

6. Click on **Next**

7. On the next step, let default enrollment policy configuration, then click on **Next**

The next step lists all Microsoft Certificate Templates on which you have enrollment rights. The Microsoft Certificate template selection and last parts of this testing procedure are specific to the enrollment mode you want to perform.

Please refer to the proper section below.

Requesting a certificate using 'Entity' enrollment mode

8. Select the Microsoft Certificate Template configured on Horizon side as a part of a **Template Mapping** using **Entity** enrollment mode. Click on **Next**

9. Click on **Enroll** to request Enrollment.

10. Enrollment is requested to WinHorizon. Few seconds later, if enrollment is successful it will be displayed **STATUS: Succeeded**. Click on **Finish**.

11. Your certificate is displayed and available.

Requesting a certificate using 'Enrollment On Behalf of Others' enrollment mode

8. Identify the Microsoft Certificate Template configured on Horizon side as a part of a **Template Mapping** using **Enrollment On Behalf of Others (EOBO)** enrollment mode. Click on **Details** then **Properties**.

9. Navigate to **Extensions** tab and select **Enrollment Agent Certificate** (to be used to sign Certificate Request). Click on **OK**.

10. Click on **Enroll** to request Enrollment.

11. Enrollment is requested to WinHorizon. Few seconds later, if enrollment is successful it will be displayed **STATUS: Succeeded**. Click on **Finish**.

12. Your certificate is displayed and available.

Requesting a certificate using 'Trust request' enrollment mode

8. Identify the Microsoft Certificate Template configured on Horizon side as a part of a **Template Mapping** using **Trust request** enrollment mode. Click on **Details** then **Properties**.

9. Navigate to **Subject** tab to build your Certificate request manually. Click on **OK**.

10. Click on **Enroll** to request Enrollment.

11. Enrollment is requested to WinHorizon. Few seconds later, if enrollment is successful it will be displayed **STATUS: Succeeded**. Click on **Finish**.

12. Your certificate is displayed and available.

8.4.8. WCCE MSAD Connector

This section details how to to configure the Microsoft Active Directory Connectors.

Uses

- WCCE Scheduled Tasks

Prerequisites

- WCCE Forest

How to configure an MSAD Connector

1. Log in to Horizon Administration Interface.

2. Access MSAD Connectors from the drawer or card: **Protocol** > **WCCE** > **MSAD Connectors**.

3. Click on  .

4. Fill in the mandatory fields.

General

- **Name*** *(select)*:
  Select the Active Directory Forrest you want to use to set up the connector.
- **Hostname*** *(string input)*:
  DNS name or IP of the Active Directory domain.
- **Port** *(string input)*:
  Port to connect to the Active Directory. The default value is set to 636.
- **Proxy***(select)*:
  Select a proxy to connect to the Active Directory, if needed.
- **Bind DN*** *(string input)*:

DN of the Active Directory account. Must have right privileges to browse and list objects.

- **Password*** *(string input)*:
  Password associated with aforementioned Active Directory DN account.
- **Timeout*** *(finite duration)*:
  The time before Horizon stop trying to connect to Active Directory. Must be in valid finite duration.
- **Max stored certificate per holder** *(int)*:
  When specified, define the maximum number of active certificates for a given Holder.

Assets identification

- **Base DN*** *(string input)*:
  It can be the root of your domain or a restriction.
- **LDAPPUB Filter** *(string input)*:
  This filter must respect LDAPPUB filter syntax.

Actor management

These configuration elements mainly define the number of authorized interactions with the remote service on a defined period. For example, one needs to ensure that the remote service will not be requested more than 5 times per 3 seconds. *Throttle parallelism* defines the number of times and *Throttle duration* the period of time. Therefore, on the above example, throttle parallelism would be set to 5 and throttle duration would be set to 3 seconds.

- **Throttle duration*** *(finite duration)*:
  The default value is set to 3 seconds. Must be a valid finite duration.
- **Throttle parallelism*** *(int)*:
  The default value is set to 3.

5. Click on the save button.

You can update  or delete  the MSAD Connector.

CAUTION	You won't be able to delete a MSAD Connector if this one is referenced somewhere else.

8.4.9. WCCE Scheduled Tasks

This section details how to schedule tasks that will run periodically on your WCCE profiles. You will be able to use MSAD Connector to browse Active Directory and retrieve changes (basically computer removal) to trigger certificate revocation. This mechanism works using comparison between Active Directory content (using MSAD connector) and Horizon certificate list based on a specific WCCE profile. If Horizon has a certificate for a holder that does not exist on Active Directory side a revocation will be triggered automatically.

Prerequisites

WCCE Forest	WCCE Profiles	MSAD Connectors

How to configure WCCE Scheduled Tasks

1. Log in to Horizon Administration Interface.

2. Access WCCE scheduled tasks from the drawer or card: **Protocol** > **WCCE** > **Scheduled Tasks**.

3. Click on  .

4. Fill the mandatory fields.

- **WCCE Profile*** *(select)*:
  Select the target WCCE profile.
- **Target Connector*** *(select)*:
  Select the MSAD connector to use as **golden source** of active Active Directory objects.
- **Cron scheduling in Quartz format** *(cron expression)*:
  Enter a Cron scheduling expression (in Quartz format). Default value is every 5 hours.
- **Revoke** *(boolean)*:
  If true, will revoke all certificate that do not exist on the AD side.
- **Dry run** *(boolean)*:
  If enabled, revocation actions will not be performed. Instead, a message will be logged, explaining what would have been done.

5. Click on the save button.

You can run  , update  or delete  the Schedules Tasks.

8.5. WebRA

8.5.1. WebRA Profile

This section details how to configure the WebRA Profile.

Required By

WebRA Scheduled Tasks

Prerequisites

WebRA Template	PKI

How to configure WebRA Profile

1. Log in to Horizon Administration Interface.

2. Access WebRA Profiles from the drawer or card: **Protocols** > **WebRA** > **Profiles**.

3. Click on  .

4. Fill in the mandatory fields.

General

- **Name*** *(string input)*:
  Enter a meaningful profile name, this setting will be the profile identifier. It must be unique for each profile.
- **Enabled** *(boolean)*:
  Should the profile be enabled. The default value is set to true.
- **WebRA Template*** *(select)*:
  Select a previously created WebRA Template.
- *PKI * *(select)*:
  Select a previously created PKI connector.
- **Max certificate per holder** *(int)*:
  When specified, define the maximum number of active certificates for a given Holder.

Cryptographic Policy

- **Allowed key type*** *(select)*:
  Select from the list the allowed key types. The default values are set to RSA / 2048 and RSA / 3072.
- **Decentralized enrollment** *(boolean)*:
  Tells whether the profile should be used with a decentralized enrollment mode, i.e CSR (PKCS#10) signing by the PKI. The default value is set to false.
- **Centralized enrollment** *(boolean)*:
  Tells whether the profile should be used with a centralized enrollment, i.e providing a PKCS#12. The default value is set to false.
  - **Private key escrowing** *(boolean)*:
    Tells whether the private key should be escrowed by Horizon. Only available if **Centralized**

enrollment is set to true. The default value is set to false.

- **PKCS#12 Password generation mode*** *(select)*:
  Define if the PKCS#12 password is chosen by the user on the request (manual) or generated randomly (random). Only available if **Centralized enrollment** is set to true.
- **Password policy for PKCS#12 password*** *(select)*:
  Select a previously created password policy. Only available if **Centralized enrollment** is set to true.
- **Store encryption type** *(select)*:
  Select from the list the encryption type. Only available if **Centralized enrollment** is set to true. The default value is set to DES_AVERAGE.
- **Show PKCS#12 Password On Enroll** *(boolean)*:
  Tells whether the PKCS#12 password should be displayed on enroll. Only available if **Centralized enrollment** is set to true. The default value is set to false.
- **Show PKCS#12 On Enroll** *(boolean)*:
  Tells whether the PKCS#12 should be displayed on enroll. Only available if **Centralized enrollment** is set to true. The default value is set to false.
- **Show PKCS#12 Password On Recover** *(boolean)*:
  Tells whether the PKCS#12 password should be displayed on recover. Enabled when the private key escrowing value is set to on. The default value is set to false.
- **Show PKCS#12 On Recover** *(boolean)*:
  Tells whether the PKCS#12 should be displayed on recover. Enabled when the private key escrowing value is set to on. The default value is set to false.

Self Permissions

- **Revoke** *(boolean)*:
  Tells whether self revoke permission is granted or not. The default value is set to false.
- **Request Revoke** *(boolean)*:
  Tells whether self request revoke permission is granted or not. The default value is set to false.
- **Update** *(boolean)*:
  Tells whether self update permission is granted or not. The default value is set to false.
- **Request Update** *(boolean)*:
  Tells whether self request update permission is granted or not. The default value is set to false.
- **Recover** *(boolean)*:
  Tells whether self recover permission is granted or not. The default value is set to false.
- **Request recover** *(boolean)*:
  Tells whether self request recover permission is granted or not. The default value is set to false.

Triggers

WebRA profiles support the use of third-party triggers in the form of callbacks on specific events happening on the profile, giving a way to synchronize the third party repositories and Horizon.

- **Enrollment** *(select)*:

Select the third party trigger(s) to call whenever a certificate is enrolled on this profile.

- **Revocation** *(select)*:
  Select the third party trigger(s) to call whenever a certificate gets revoked on this profile.
- **Expire** *(select)*:
  Select the third party trigger(s) to call whenever a certificate expires on this profile.

You can further configure the profile using the Common configuration profile and Notification tabs.

5. Click on the save button.

You can edit  , duplicate  or delete  the WebRA Profile.

CAUTION	You won't be able to delete a WebRA Profile if it is referenced somewhere else.

8.5.2. WebRA Scheduled Tasks

This section details how to schedule tasks that will run periodically with your WebRA profiles.

Prerequisites

AWS Connector	AKV Connector	F5 Connector	WebRA Profile

How to configure WebRA Scheduled Tasks

1. Log in to Horizon Administration Interface.

2. Access the "Scheduled tasks" from the drawer or card: **Protocols** > **WebRA** > **Scheduled Tasks**.

3. Click on  .

4. Fill in the mandatory fields.

- **WebRA Profile*** *(select)*:
  Select a previously created WebRA profile.
- **Target Connector*** *(select)*:
  Select a previously created third party connector.
- **Cron scheduling** *(cron expression)*:
  Enter a Cron scheduling expression (in Quartz format). The default expression is built to run the task every 5 hours.
- **Revoke** *(boolean)*:
  If enabled, will revoke all certificate whose container was deleted from the third party repository. The default value is set to false.
- **Renew** *(boolean)*:
  If enabled, will renew all certificate who are about to expire. The default value is set to false.
- **Dry run** *(boolean)*:
  If enabled, revocation and renewal actions will not be performed. Instead, a message will be logged, explaining what would have been done.

5. Click on the save button.

You can run  or edit  or delete  the Schedules Tasks.

8.5.3. WebRA Template

This section details how to define a custom structure for the fields `subject DN`& `SAN` of the requested certificate in order to match the configuration on the PKI side.

How to configure WebRA Template

1. Log in to Horizon Administration Interface.

2. Access WebRA Templates from the drawer or card: **Protocols** > **WebRA** > **Templates**.

3. Click on  .

4. Fill in the mandatory fields.

Details

- **Template Name*** *(string input)*: Enter a meaningful WebRA template name. It must be unique for each template.

Subject DN composition

You can add more elements by clicking the add button.

- **Element*** *(select)*:
  Select an attribute from the elements list.
- **Mandatory** *(boolean)*:
  Should the element be mandatory. The default value is set to false.
- **Default value** *(string input)*:
  Set a default value to the element.
- **Regex** *(string input)*:
  Enter a regular expression that the element should match.
- **Editable by requester** *(boolean)*:
  Tells whether the element should be editable by the requester. The default value is set to false.
- **Editable by approver** *(boolean)*:
  Tells whether the element should be editable by the approver. The default value is set to false.

You can remove an element by clicking the delete button  .

SAN composition

You can add more elements by clicking the add button.

- **Element*** *(select)*:
  Select an attribute from the element list.
- **Mandatory** *(boolean)*:

Tells whether the element should be mandatory. The default value is set to false.

- **Default value** *(string input)*:
  Set a default value to the element.
- **Regex** *(string input)*:
  Enter a regular expression that the element should match.
- **Editable by requester** *(boolean)*:
  Tells whether the element should be editable by the requester. The default value is set to false.
- **Editable by approver** [.mysize]_(boolean)_s:
  Tells whether the element should be editable by the approver. The default value is set to false.

You can remove an element by clicking the delete button  .

NOTE	When adding a SAN or a DN element and making it mandatory, make sure to either give it a default value or make it editable by the requester, otherwise the template will be unusable.

5. Click on the save button.

You can edit  , duplicate  or delete  the WebRA template.

CAUTION	You won't be able to delete WebRA template if it is referenced somewhere else.

8.6. Common configuration elements for profiles

This section details how to configure sections that are common to all profiles

Common configuration for profiles tab

8.6.1. Common configuration for profiles

Languages

1. Click on add button.

2. Fill in the mandatory fields.

- **Language*** *(string input)*:
  Select a language.
- **Display Name*** *(string input)*:
  Enter a display name.
- **Description** *(string input)*:
  Enter a description.

You can add more by clicking on the add button again or delete  the language.

Labels

1. Click on add button.

2. Fill in the mandatory fields.

- **Element*** *(string input)*:
  Select a preexisting label.
- **Mandatory** *(boolean)*:
  Tells whether the label is mandatory. The default value is set to false.
- **Editable by requester** *(boolean)*:
  Tells whether the label is editable by the requester. The default value is set to false.
- **Editable by approver** *(boolean)*:
  Tells whether the label is editable by the approver. The default value is set to false.
- **Default value** *(string input)*:
  Set a default value to the label. This value must comply with the value restriction.
- **Label value restriction**
  - **Whitelist** *(string input multiple)*:
    The label value will have to be in the whitelist. Enter the label value and press "enter" to add this value to the accepted value list.
  - **Regex** *(string input)*:
    The label value will have to match the regex. Enter the regular expression en click on the check button to set the regex.

You can delete  or reorder (drag and drop)  the label template.

Owner Policy

1. Specify the request's owner policy (only used in EST, SCEP and WEBRA prevalidated request).

- **Editable by requester** *(boolean)*:
  Specify if the certificate's owner can be overridden by the requester when submitting a request.
- **Editable by approver** *(boolean)*:
  Specify if the certificate's owner can be overridden by the requester when approving a request.

Team Policy

1. Specify the request's team policy (only used in EST, SCEP and WebRA prevalidated request).

- **Editable by requester** *(boolean)*:
  Specify if the certificate's team can be overridden by the requester when submitting a request.
- **Editable by approver** *(boolean)*:
  Specify if the certificate's team can be overridden by the requester when approving a request.
- **Team restriction**

    - **Whitelist** *(string input multiple)*:
      The team will have to be in the whitelist. Enter the team and press "enter" to add this value to the accepted whitelist.
    - **Regex** *(string input)*:
      The team will have to match the regex. Enter the regular expression and click on the check button to set the regex.

- **Default team** *(string input)*:
  Set a default team. This value must comply with the team restriction.

Metadata policy *(overridable metadata)*

WARNING	== Metadata are used by Horizon or Third party connectors, updating them should be done with utmost care. ==

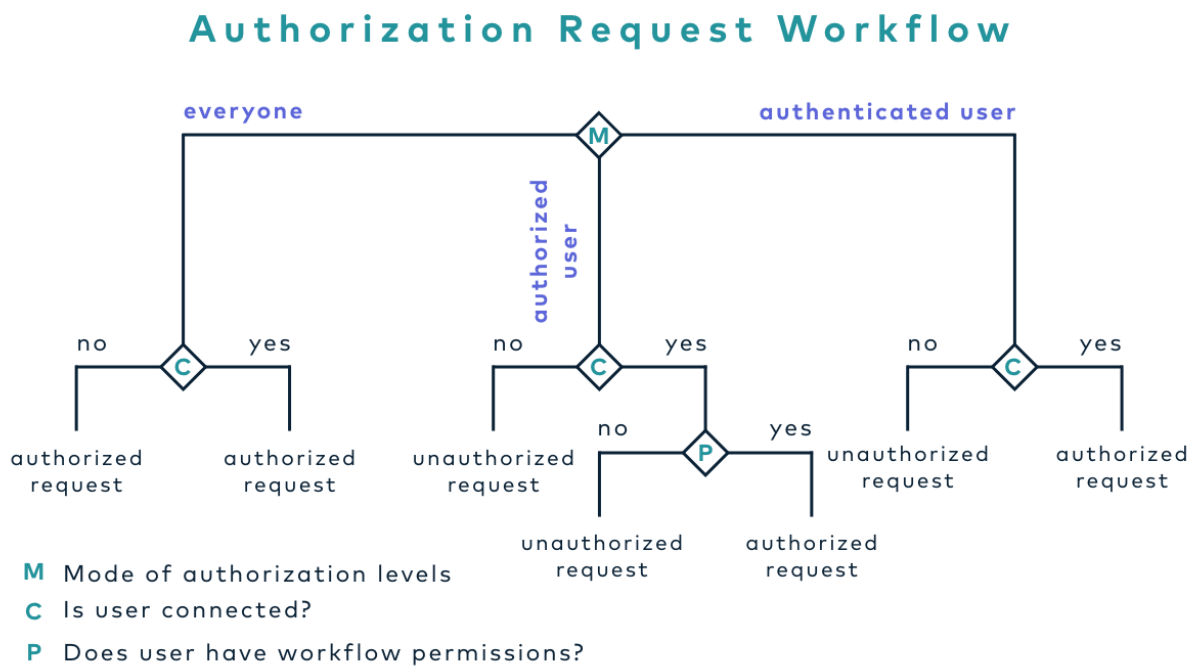
NOTE	== The contact email metadata default value is set to editable by the requester and the approver. ==

1. Click on add button.

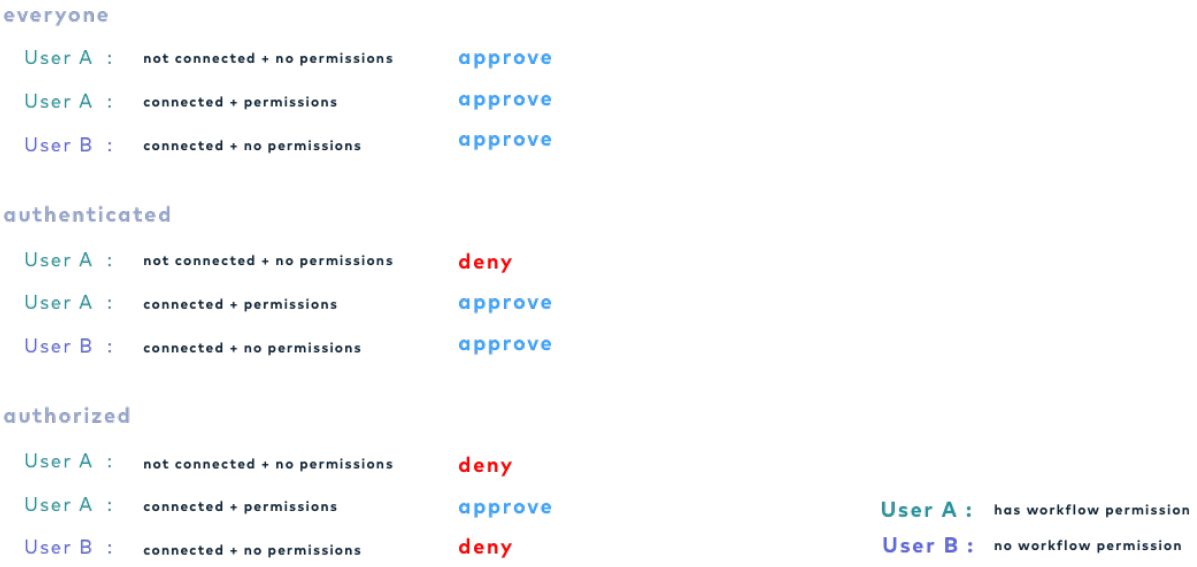
- **Metadata*** *(select)*:
  Select a metadata.
- **Editable by requester** *(boolean)*:
  Tells whether the metadata is editable by the requester. The default value is set to false.
- **Editable by approver** *(boolean)*:
  Tells whether the metadata is editable by the approver. The default value is set to false.

Authorization Levels

1. Select an authorization level for each workflow.



- *Everyone:
  No authentication is required.
- **Authenticated**:
  User has to be authenticated.
- **Authorized**:
  User has to be authenticated and have an explicit authorizations.



2. Select an access level for identity providers.

You can remove the access level for an identity provider by clicking on 'x'.

Requests time to live (TTL)

1. Enter a time for each request.

- **Enrollment request*** *(finite duration)*:
  Must be in valid finite duration. The default value is set to one hour.
- **Revocation request*** *(finite duration)*:
  Must be in valid finite duration. The default value is set to one hour.
- **Update request*** *(finite duration)*:
  Must be in valid finite duration. The default value is set to one hour.
- **Migration request*** *(finite duration)*:
  Must be in valid finite duration. The default value is set to one hour.
- **Recover request** *(finite duration)*:
  Must be in valid finite duration. This field is enabled when Private key escrowing is set to on (Specfic configuration tab > Crypto Policy).

Constraints

ACME, EST, SCEP and WCCE protocols.

1. Fill in the mandatory fields.

- **RSA Minimal Key size** *(select)*:
  Select the allowed RSA key size(s).
- **Allowed EC curves** *(select)*:
  Select the allowed elliptic curve algorithms.
- **Allowed email domains** *(string input)*:
  Enter a valid regular expression that the inputted domain should match.
- **Allowed DNS domains** *(string input)*:
  Enter a valid regular expression that the inputted domain should match.

CSR Data Mapping

1. Click on add button.

2. Select a field and enter a value.

You can add more by clicking on the add button again or delete  the CSR Data Mapping.

8.6.2. Notification

Notifications configuration tab

8.6.3. Prerequisites

Mail, Messaging

8.6.4. How to manage Notification for profiles

Certificate lifecycle notifications

Notifications are sent when one of the following event is triggered by a certificate:

Enrollment	Revocation	Expire	Update	Migrate

1. Chose notifications to be sent on certificate event.

Request lifecycle notifications

Notifications are sent when one of the following event is triggered by a request:

- Enroll/Revocation/Update/Migrate request :

Submit	Cancel	Revoke	Approve	Pending

You can delete  the Notification for profile.

9. Third parties

9.1. AWS Certificate Manager Integration

9.1.1. Introduction

This section refers to the AWS Certificate Manager (ACM) integration with Horizon, used to enroll certificates held in ACM.

This integration involves at least two infrastructure components:

- AWS Certificate Manager
- EverTrust Horizon

9.1.2. AWS Connector

Here is the section to manage the AWS Connector.

Required By

- AWS Trigger

Prerequisites

On Horizon side, you might need to set up a Proxy , used to reach AWS, if necessary.

On AWS side, you need to create a user using the AWS IAM module, and following AWS guide. You should create an access key for that user, and give him appropriate permissions. The created user should hold the following permissions:

- `AWSResourceGroupsReadOnlyAccess`
- `ResourceGroupsandTagEditorReadOnlyAccess`
- `AWSCertificateManagerFullAccess`

After performing these steps, you will get the following information, required later:

- the AWS Region
- the User Access Key ID
- the User Access Key Secret

On top of that, you need to define a Resource Group, using AWS Resource Groups and Tags Editor, with the following characteristics:

- Group Type: Tag based
- Resource Type: `AWS::CertificateManager::Certificate`

- Tag key and value (e.g. key=manage and value=HRZ)

After performing this steps, you will get the following information, required later:

- The Resource Group name
- the Tag name
- the Tag value

How to configure AWS Connector

1. Log in to Horizon Administration Interface.

2. Access AWS Connectors from the drawer or card: **Third Parties** > **AWS** > **Connectors**.

3. Click on  .

4. Fill the mandatory fields.

Connection

- **Name*** *(string input)*:
  Enter a meaningful connector name. It must be unique for each connector. Horizon uses the name to identify the connector.
- **Region*** *(string input)*:
  Enter a valid AWS region. Here's the region list from AWS.
- **Access key ID** *(string input)*:
  User Access Key ID used by Horizon to connect to AWS.
- **Access Key Secret** *(string input)*:
  Access Key Secret associated to the aforementioned User Access Key ID.
- Proxy *(select)*:
  The HTTP/HTTPS proxy to use to reach AWS, if any.
- **Timeout*** *(finite duration)*:
  The timeout for Horizon-initiated connections to AWS. Must be a valid finite duration.

Assets identification

- **Resource group name** *(string input)*:
  Name of the resource group pointing to the tag name and value.
- **Tag key** *(string input)*:
  Name of the tag used to identify certificates managed by Horizon in ACM.
- **Tag value** *(string input)*:
  Value of the tag used to identify certificates managed by Horizon in ACM.

Actors and renewal management

- **Throttle duration*** *(finite duration)*:
  Set by default at 3 seconds. Must be in valid finite duration.
- **Renewal period** *(finite duration)*:
  Certificate renewal period (time before expiration to trigger renewal). Must be in valid finite duration.

5. Click on the save button.

You can update  or delete  the AWS Connector.

CAUTION	You won't be able to delete an AWS Connector if it is referenced somewhere else.

9.1.3. AWS Trigger

Here is the section to manage the Triggers that will be used by WebRA Profiles to push or delete certificates to/from AWS ACM.

Prerequisites

AWS Connector

How to configure AWS Trigger

1. Log in to Horizon Administration Interface.

2. Access AWS Triggers from the drawer or card: **Third Parties** > **AWS** > **Triggers**.

3. Click on  .

4. Fill the mandatory fields.

- **Name*** *(string input)*:
  Enter a meaningful trigger name. It must be unique for each trigger. Horizon uses the name to identify the trigger.
- **AWS Connector*** *(select)*:
  Select an AWS connector previously created.
- **Retries in case of error** *(int)*:
  Number of times to retry to push the change on the AWS repository in case of error. Must be an integer between 1 and 15.

5. Click on the save button.

You can update  or delete  the AWS Trigger.

CAUTION	You won’t be able to delete an AWS Trigger if it is referenced somewhere else.

9.1.4. Integration of the third party to the WebRA

When having configured the connector, it is possible to automate its elements' lifecycle using the WebRA.

Automation using triggers

Triggers are a functionality of WebRA that allows to push lifecycle events into a third party whenever they occur on a WebRA profile.

1. Refer to the trigger documentation to create a trigger.

2. Create or modify the WebRA profile you wish to use the triggers on.

3. Go to the **Triggers** tab, then on **Certificate lifecycle triggers**

4. Chose which lifecycle events you wish to use triggers upon (enrollment, revocation, expiration)

5. Select one or more existing triggers from the menu (if several are selected, they will all be called whenever the selected event occurs)

6. Click on the Save button.

From now on, whenever a selected lifecycle event will occur on the configured WebRA profile, the trigger will be called and the and the certificate will be pushed into or removed from the third party container.

Automation using scheduled tasks

Scheduled tasks are a functionality of WebRA that allows to synchronize automatic renewal or revocation events with a third party periodically with what occurs on a WebRA profile. To be more specific, it will periodically check whether the certificate has entered the "renewal period" that was defined in the connector’s configuration, and renew it automatically if necessary.

1. Refer to the third party connector documentation to create a third party connector.

2. Ensure you have an existing WebRA profile : renewal will be automated on the selected profile.

3. Follow the documentation of the WebRA scheduled tasks section to properly configure a scheduled task.

9.2. Azure Key Vault Integration

9.2.1. Introduction

This section refers to the Azure Key Vault (AKV) integration with Horizon, used to enroll certificates held in AKV.

This integration involves at least three infrastructure components:

- Azure Key Vault
- Azure Active Directory
- EverTrust Horizon

Azure AD is used to authenticate Horizon, which should be a registered application.

9.2.2. Azure AKV Connector

Here is the section to manage the Azure AKV Connector.

Required By

AKV Trigger

Prerequisites

On Horizon side, you might need to set up a Proxy used to reach Azure, if necessary.

On Azure AD side, it is required to set up an application by following Microsoft's guide.

NOTE	Horizon supports only client secret authentication

After performing these steps, you will get the following information, required later:

- the Tenant ID
- the Application ID
- the Application Authentication Key

Finally, you should give all Certificate Permissions to the Application you created for Horizon inside the target Azure Key Vault "Access policies" menu entry, using the "Add Access Policy" link.

How to configure AKV Connector

1. Log in to Horizon Administration Interface.

2. Access AKV Connectors from the drawer or card: **Third Parties** > **AKV** > **Connectors**.

3. Click on  .

4. Fill the mandatory fields.

Connection

- **Name*** *(string input)*:
  Enter a meaningful Connector Name.

- **Azure Tenant*** *(string input)*:
  Enter the Tenant, which is the domain name after the @ sign in your account.
- **App ID*** *(string input)*:
  Enter the app ID.
- **App Key*** *(string input)*:
  Enter the app Key.
- Proxy *(select)*:
  The HTTP/HTTPS proxy used to reach Azure AD and AKV, if necessary.
- **Timeout** *(finite duration)*:
  Set on the connections used to reach Azure AD and AKV. Configured by default at 10 seconds. Must be in valid finite duration.
- **Vault fully qualified domain name*** *(string input)*:
  Fully qualified domain name used to reach the Azure Key Vault to be managed by Horizon.

Assets identification and management

- **Prefix** *(string input)*: Used to filter the certificates managed by Horizon in the specified Azure Key Vault. Defaults to "HRZ-"

Actors and renewal management

These configuration elements mainly define the number of authorized interactions with the remote service on a defined period. For example, one needs to ensure that the remote service will not be contacted more than 5 times per 3 seconds. *Throttle parallelism* defines the number of times and *Throttle duration* the period of time. Therefore, on the above example, throttle parallelism would be set to 5 and throttle duration would be set to 3 seconds.

- **Throttle duration*** *(finite duration)*:
  Set by default at 3 seconds. Must be in valid finite duration.
- **Throttle parallelism*** *(int)*:
  Set by default at 3.
- **Renewal period** *(finite duration)*:
  Must be a valid finite duration.

5. Click on the save button.

You can update  or delete  the AKV Connector.

CAUTION	You will not be able to delete an AKV Connector if it is referenced in any other configuration element.

9.2.3. AKV Trigger

This section details how to configure the Triggers that will be used by WebRA Profiles to push or delete certificates to/from AKV.

Prerequisites

AKV Connector

How to configure AKV Trigger

1. Log in to Horizon Administration Interface.

2. Access AKV Triggers from the drawer or card: **Third Parties** > **AKV** > **Triggers**.

3. Click on  .

4. Fill the mandatory fields.

- **Name*** *(string input)*:
  Enter a meaningful trigger name. It must be unique for each trigger. Horizon uses the name to identify the trigger.
- **Azure Key Vault Connector*** *(select)*:
  Select an AKV connector previously created.
- **Retries in case of error** *(int)*:
  Number of times to retry to push the change on the AKV repository in case of error. Must be an integer between 1 and 15.

5. Click on the save button.

You can update  or delete  the AKV Trigger.

9.2.4. Integration of the third party to the WebRA

When having configured the connector, it is possible to automate its elements' lifecycle using the WebRA.

Automation using triggers

Triggers are a functionality of WebRA that allows to push lifecycle events into a third party whenever they occur on a WebRA profile.

1. Refer to the trigger documentation to create a trigger.

2. Create or modify the WebRA profile you wish to use the triggers on.

3. Go to the **Triggers** tab, then on **Certificate lifecycle triggers**

4. Chose which lifecycle events you wish to use triggers upon (enrollment, revocation, expiration)

5. Select one or more existing triggers from the menu (if several are selected, they will all be called whenever the selected event occurs)

6. Click on the Save button.

From now on, whenever a selected lifecycle event will occur on the configured WebRA profile, the trigger will be called and the and the certificate will be pushed into or removed from the third party container.

Automation using scheduled tasks

Scheduled tasks are a functionality of WebRA that allows to synchronize automatic renewal or revocation events with a third party periodically with what occurs on a WebRA profile. To be more specific, it will periodically check whether the certificate has entered the "renewal period" that was defined in the connector's configuration, and renew it automatically if necessary.

1. Refer to the third party connector documentation to create a third party connector.

2. Ensure you have an existing WebRA profile : renewal will be automated on the selected profile.

3. Follow the documentation of the WebRA scheduled tasks section to properly configure a scheduled task.

9.3. F5 BigIP Integration

9.3.1. Introduction

This section refers to the F5 BigIP integration with Horizon, used to enroll certificates used by F5 BigIP.

This integration involves at least two infrastructure components:

- F5 BigIP
- EverTrust Horizon

Horizon connects to the F5 BigIP using the iControl REST administration API in order to manage the lifecycle of certificates associated to Client SSL Profiles within the BigIP.

9.3.2. F5 Connector

This section details how to configure the F5 Connector.

Prerequisites

On the F5 BigIP side, you need to create a technical user for Horizon, and give it full administrator rights. This is required because only full admins have the right to upload certificates on an F5 BigIP.

After performing these steps, you will get the following information, required later:

- the technical user login/username
- the technical user password

How to configure F5 Connector

1. Log in to Horizon Administration Interface.

2. Access F5 Connectors from the drawer or card: **Third Parties** > **F5** > **Connectors**.

3. Click on  .

4. Fill the mandatory fields.

General

- **Name*** *(string input)*:
  Enter a meaningful connector name. It must be unique for each connector. Horizon uses the name to identify the connector.
- **F5 BigIP hostname*** *(string input)*:
  Enter the F5 BigIP hostname (DNS or IP address).
- **F5 BigIP username*** *(string input)*:
  Username created for Horizon in the F5 BigIP. Must have administrator rights.
- **F5 BigIP password*** *(string input)*:
  Password associated with aforementioned username.
- Proxy *(select)*:
  The HTTP/HTTPS proxy to use.
- **Timeout** *(finite duration)*:
  Set by default at 10 seconds. Must be a valid finite duration*.
- **Max stored certificates per holder** *(int)*:
  When specified, define the maximum number of certificates stored in the third party for a given holder.

Assets identification

- **Partition** *(string input)*:
  F5 BigIP partition to manage. *Common* by default.
- **SSL parent** *(string input)*:
  Name of the parent Client SSL Profile. *Common* by default.
- **Prefix** *(string input)*:
  Used to filter the certificates managed by Horizon in the specified F5 Client. *hrz-* by default.
- **Cipher group** *(string input)*:
  Name of the Cipher group. *None* by default.

Actors and renewal management

These configuration elements mainly define the number of authorized interactions with the remote service on a defined period. For example, one needs to ensure that the remote service will not be contacted more than 5 times per 3 seconds. *Throttle parallelism* defines the number of times and *Throttle duration* the period of time. Therefore, on the above example, throttle parallelism would be set to 5 and throttle duration would be set to 3 seconds.

- **Throttle duration*** *(finite duration)*:
  Set by default at 3 seconds. Must be in valid finite duration*.
- **Throttle parallelism*** *(int)*:
  Set by default at 3.
- **Renewal period*** *(finite duration)*:
  Must be a valid finite duration*.

5. Click on the save button.

You can update  or delete  the F5 Connector.

CAUTION	You will not be able to delete an F5 Connector if it is referenced in any other configuration element.

9.3.3. F5 Trigger

This section details how to configure the Triggers that will be used by WebRA Profiles to push or delete certificates to/from F5 BigIP.

Prerequisites

F5 Connector

How to configure F5 Trigger

1. Log in to Horizon Administration Interface.

2. Access F5 Triggers from the drawer or card: **Third Parties** > **F5** > **Triggers**.

3. Click on  .

4. Fill the mandatory fields.

- **Name*** *(string input)*:
  Enter a meaningful trigger name. It must be unique for each trigger. Horizon use the name to identify the trigger.
- **F5 Connector*** *(select)*:
  Select a connector F5 previously created.
- **Retries in case of error** *(int)*:
  Number of times to retry to push the change on the F5 BigIP repository in case of error. Must be an integer between 1 and 15.

5. Click on the save button.

You can update  or delete  the F5 Trigger.

9.3.4. Integration of the third party to the WebRA

When having configured the connector, it is possible to automate its elements' lifecycle using the WebRA.

Automation using triggers

Triggers are a functionality of WebRA that allows to push lifecycle events into a third party whenever they occur on a WebRA profile.

1. Refer to the trigger documentation to create a trigger.

2. Create or modify the WebRA profile you wish to use the triggers on.

3. Go to the **Triggers** tab, then on **Certificate lifecycle triggers**

4. Chose which lifecycle events you wish to use triggers upon (enrollment, revocation, expiration)

5. Select one or more existing triggers from the menu (if several are selected, they will all be called whenever the selected event occurs)

6. Click on the Save button.

From now on, whenever a selected lifecycle event will occur on the configured WebRA profile, the trigger will be called and the and the certificate will be pushed into or removed from the third party container.

Automation using scheduled tasks

Scheduled tasks are a functionality of WebRA that allows to synchronize automatic renewal or revocation events with a third party periodically with what occurs on a WebRA profile. To be more specific, it will periodically check whether the certificate has entered the "renewal period" that was defined in the connector's configuration, and renew it automatically if necessary.

1. Refer to the third party connector documentation to create a third party connector.

2. Ensure you have an existing WebRA profile : renewal will be automated on the selected profile.

3. Follow the documentation of the WebRA scheduled tasks section to properly configure a scheduled task.

9.4. Intune

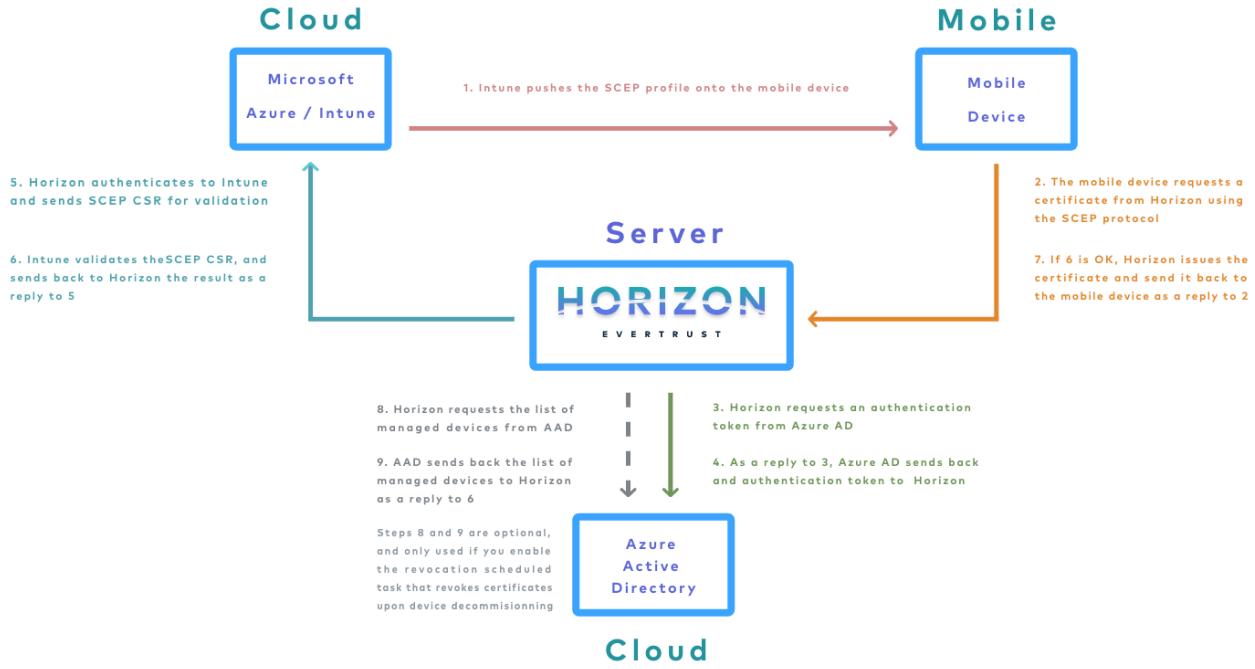
9.4.1. Introduction

This section details the Microsoft Endpoint Manager - Intune SCEP integration with Horizon, used to enroll, renew and revoke certificates on Intune managed devices.

This integration involves at least three infrastructure components:

- Microsoft Endpoint Manager / Intune
- Azure Active Directory
- EverTrust Horizon

The enrolled devices interface with these components in order to retrieve their certificate.



The diagram displays these components as well as the various flows involved in an enrollment.

Microsoft describes the integration principles on their website: https://docs.microsoft.com/en-us/mem/intune/protect/certificate-authority-add-scep-overview

Finally, this integration will require to set up, on Horizon side, the following elements:

- an Intune Connector, which holds the configuration items required for Horizon to connect to Azure AD and Intune
- an Intune Profile, which holds the configuration items specifying how Horizon should issue certificates for the specified Intune Connector
- an Intune Scheduled Task, which holds configuration items defining the scheduled task in charge of performing revocation upon decommissioning devices from Azure AD. This is optional.

9.4.2. Intune Connector

This section details how to configure an Intune Connector.

Required By

Intune Profile	Intune Scheduled Task

Prerequisites

On Horizon side, you might need to set up a Proxy, used to reach Azure/Intune, if necessary.

On Azure AD side, it is required to set up an application by following Microsoft's guide. Please note that you must add the **Microsoft Graph / Directory.Read.All** permission as well for the revocation

feature to work properly. After performing these steps, you will get the following information, required later:

- the Tenant ID
- the Application ID
- the Application Authentication Key

How to configure Intune Connector

1. Log in to Horizon Administration Interface.

2. Access Intune Connector from the drawer or card: **Third Parties** > **Intune** > **Connectors**.

3. Click on  .

4. Fill the mandatory fields.

Connection

- **Name*** *(string input)*:
  Enter a meaningful connector name. It must be unique for each connector. Horizon uses the name to identify the connector.
- **Azure Tenant*** *(string input)*:
  Enter the Tenant ID.
- **App ID*** *(string input)*:
  Enter the Application ID.
- **App Key*** *(string input)*:
  Enter the Application Authentication Key.
- Proxy *(select)*:
  The HTTP/HTTPS proxy used to reach Azure AD and Intune.
- **Timeout** *(finite duration)*:
  Timeout set on the connection used to reach Azure AD and Intune. Configured by default at 10 seconds. Must be a valid finite duration.

Assets identification and management

- **OS query string** *(string input)*:
  This allows to restrict devices by OS when performing the devices listing used for the revocation feature. Leave blank to use the default setting if unsure.
- **Intune resource URL** *(string input)*:
  This allows to point at a specific Intune installation. Used only in Hybrid Intune setups, leave blank otherwise.
- **Legacy revocation mode** *(boolean)*:
  Activate the legacy revocation mode. Default value is set to false.

Actors management

These configuration elements mainly define the number of authorized interactions with the remote service on a defined period. For example, one needs to ensure that the remote service will not be contacted more than 5 times per 3 seconds. *Throttle parallelism* defines the number of times and *Throttle duration* the period of time. Therefore, on the above example, throttle parallelism would be set to 5 and throttle duration would be set to 3 seconds.

- **Throttle duration*** *(finite duration)*:
  Set by default to 3 seconds. Must be a valid finite duration.
- **Throttle parallelism*** *(int)*:
  Set by default to 3.

5. Click on the save button.

You can update  or delete  the Intune Connector.

CAUTION	You will not be able to delete an Intune Connector if it is referenced in any other configuration element.

9.4.3. Intune Profile

This section details how to configure an Intune Profile.

Required By

- Intune Scheduled Tasks

Prerequisites

Intune Connector	PKI Connector	SCEP Authority

Setting up an SCEP Authority requires you to issue a certificate from the underlying PKI with the following characteristics:

- the issuing CA should be the same as the one that will issue certificates through the PKI Connector that will be linked to the Intune Profile
- the certificate key usages must include **Digital Signature** and **Key Encipherment**
- the certificate must be issued as PKCS#12 and then imported into Horizon

How to configure Intune Profile

1. Log in to Horizon Administration Interface.

2. Access Intune Profile from the drawer or card: **Third Parties** > **Intune** > **Profiles**.

3. Click on  .

4. Fill the mandatory fields.

General

- **Name*** *(string input)*:
  Enter a meaningful connector name. It must be unique for each profile. Horizon uses the name to identify the profile. As the name will be part of an URL, it is advised to use only lower case letters and dashes.
- **Enabled*** *(boolean)*:
  Indicates whether the profile is enabled or not. Set to true by default.
- **Intune Connector*** *(select)*:
  Select an Intune Connector previously created.
- **PKI Connector** *(select)*:
  Select a PKI connector previously created.
- **Max certificate per holder** *(int)*:
  If specified, defines the maximum number of active certificates for a given Holder. If the number of active certificates exceeds this parameter, then the oldest certificate(s) above the limit will be automatically revoked.

Assets identification

- **Device ID field name** *(string input)*:
  Subject DN field used to retrieve the Device ID. The selected field must be set to `{{AAD_Device_ID}}` on Intune side, e.g. if you select "L", the configured Subject DN in the SCEP profile in Intune must then contain `L={{AAD_Device_ID}}`. This is required to use the automated revocation feature upon device decommission.
- **Device ID separator** *(string input)*:
  Separator used to retrieve the Device ID in the device id field (if defined). This field is present for backward compatibility reasons and should normally be left to blank.

SCEP protocol parameters

- **Mode*** *(select)*:
  Choose from one of the two modes RA or CA. Usually this should be set to **RA**.
- **SCEP Authority** *(select)*:
  Select a SCEP Authority previously created. See Prerequisites for details.
- **CAPS** *(select)*:
  Select one or many SCEP Capabilities from the list. If unsure, leave the default.
- **Encryption algorithm** *(select)*:
  Select a SCEP Encryption Algorithm algorithms from the list. If unsure, leave the default.

Renewal management

- **Renewal period** *(finite duration)*:
  Must be a valid finite duration.
- **Revocation on SCEP renew***: *(boolean)*
  Should the expiring certificate be revoked upon SCEP renewal. Set by default to false.
- **Revocation reason*** *(select)*:
  Select the reason from the list. Available only if revocation on SCEP renew is set to true.

Self Permissions

- **Revoke** *(boolean)*:
  Specify whether the certificate's owner is authorized to revoke the certificate with no validation workflow. Set to false by default.
- **Request Revoke** *(boolean)*:
  Specify whether the certificate's owner is authorized to request the revocation of the certificate. Set to false by default.
- **Update** *(boolean)*:
  Specify whether the certificate's owner is authorized to update the certificate with no validation workflow. Set to false by default.
- **Request Update** *(boolean)*:
  Specify whether the certificate's owner is authorized to request certificate's update. Set to false by default.

You can further configure the profile using the Common configuration profile and Notification tabs.

5. Click on the save button.

You can update  or delete  the Intune Profile once it has been created.

CAUTION	You won't be able to delete an Intune Profile if it is referenced somewhere else.

Last steps

Once the profile created in Horizon, you need to setup a SCEP profile in Intune by following Microsoft documentation. You will need to match the parameters in the Intune SCEP profile with what has been set up in Horizon and in the underlying PKI. You need to pay special attention to:

- the certificate lifetime and renewal interval, which must match throughout the solution
- the Subject and Subject Alternative Name settings must match throughout the solution. In the end, the issued certificate must contain exactly what was configured in Intune for these fields, or the renewal will not work.
- the SCEP server URL, where you need to input the URL given in the Intune Profile that you created in Horizon

Configuration settings Edit

Certificate type	User
Subject name format	CN={{UserName}}-ios,OU=Mobile,L={{AAD_Device_ID}},O=EverTrust,C=FR
Subject alternative name	

Attribute	Value
Email address	{{EmailAddress}}
User principal name (UPN)	{{UserPrincipalName}}

Certificate validity period	2 Days
Key usage	Key encipherment, Digital signature
Key size (bits)	2048
Root Certificate	EVTQA-RootCA-iOS
Extended key usage	

Name	Object Identifier	Predefined values
Client Authentication	1.3.6.1.5.5.7.3.2	Client Authentication (1.3.6.1...

Renewal threshold (%)	98
SCEP Server URLs	https://horizon-demo.evertrust.fr/intune/evertrustqa-intune/pkiclient.exe

CAUTION	To enroll **Windows** machines or users using Intune, you need to remove the trailing " `pkiclient.exe` " from the SCEP server URL

9.4.4. Intune Scheduled Tasks

This section details how to configure scheduled tasks which will run periodically on your Intune profiles, in order to manage automatic revocation upon device decommission.

Prerequisites

Intune Connector	Intune Profile

How to configure Intune Scheduled Tasks

1. Log in to Horizon Administration Interface.

2. Access Intune Scheduled Tasks from the drawer or card: **Third Parties** > **Intune** > **Scheduled Tasks**.

3. Click on  .

4. Fill the mandatory fields.

- **Intune Profile*** *(select)*:
  Select an Intune profile previously created.
- **Target Connector*** *(select)*:
  Select an Intune connector previously created.
- **Cron scheduling** *(cron expression)*:
  Set to every 5 hours by default.
- **Revoke** *(boolean)*:
  Set to false by default. If true, Horizon will revoke any certificate associated to a device that has been deleted from Azure AD (and hence decommissioned).
- **Dry run** *(boolean)*:
  If enabled, revocation actions will not be performed. Instead, a message will be logged, explaining what would have been done.

5. Click on the save button.

You can run  , update  or delete  the Scheduled Tasks.

9.5. Intune PKCS Connector

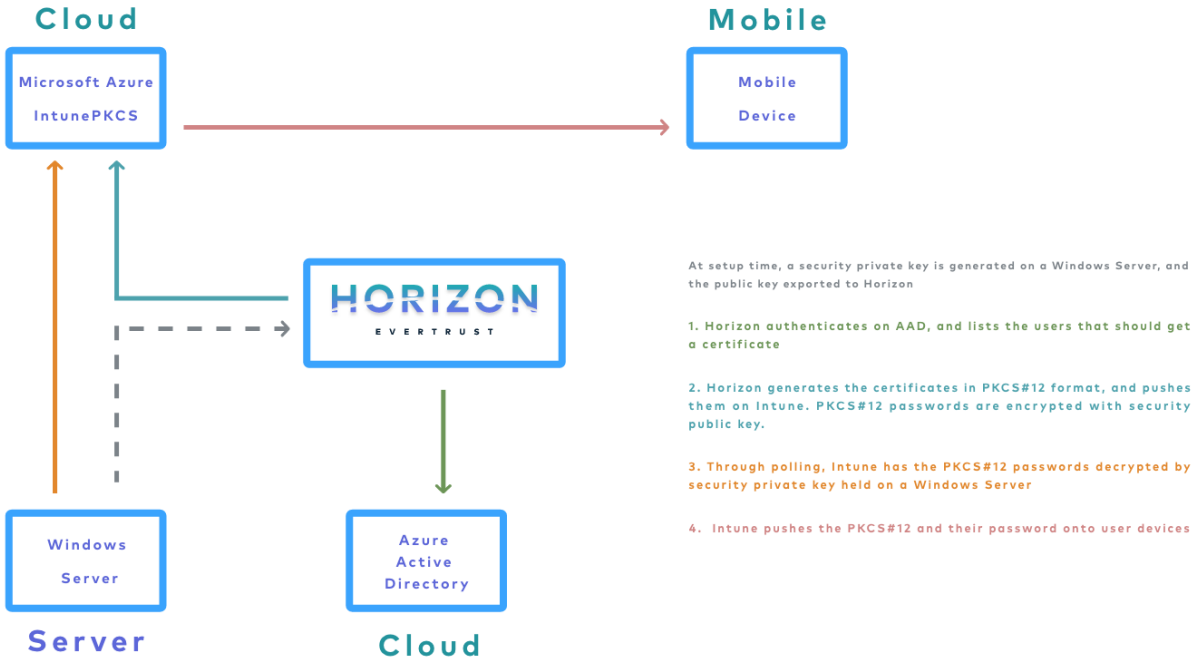
This section details how to configure the Intune PKCS Connector.

This integration involves at least three infrastructure components:

- Microsoft Endpoint Manager / Intune
- Azure Active Directory

- EverTrust Horizon

The enrolled devices interface with these components in order to retrieve their certificate.



The diagram displays these components as well as the various flows involved in an enrollment.

9.5.1. Required By

Intune PKCS Profile

9.5.2. How to configure Intune PKCS Connector

1. Log in to Horizon Administration Interface.

2. Access Intune PKCS Connectors from the drawer or card: **Third Parties** > **Intune PKCS** > **Connectors**.

3. Click on ⊕ .

4. Fill the mandatory fields.

General

- **Name*** *(string input)*:
  Enter a meaningful connector name. It must be unique for each connector. Horizon uses the name to identify the connector.
- **Azure Tenant*** *(string input)*:
  Value must be set to Azure Tenant.
- **App ID*** *(string input)*:

Value must be set to Azure App ID.

- **App Key*** *(string input)*:
  Value must be set to Azure App Key.
- Proxy *(select)*:
  The HTTP/HTTPS proxy to use.
- **Timeout** *(finite duration)*:
  Set by default at 10 seconds. Must be in valid finite duration.
- **Search Filter** *(string input)*:
  Enter search filter.
- **Max stored certificates per holder** *(int)*:
  When specified, define the maximum number of certificates stored in the third party for a given holder.

Assets identification and management

- **Key Name** *(string input)*:
  Enter key name.
- **Key Type** *(select)*:
  Select one key type from the list.
- **Provider Name** *(string input)*:
  Enter provider name.
- **Public Key** *(string input)*:
  Enter public key in PEM format.
- **Intended Purpose** *(select)*:
  Select one intended certificate usage from the list.

Actors and renewal management

These configuration elements mainly define the number of authorized interactions with the remote service on a defined period. For example, one needs to ensure that the remote service will not be contacted more than 5 times per 3 seconds. *Throttle parallelism* defines the number of times and *Throttle duration* the period of time. Therefore, on the above example, throttle parallelism would be set to 5 and throttle duration would be set to 3 seconds.

- **Throttle duration*** *(finite duration)*:
  Set by default at 3 seconds. Must be a valid finite duration.
- **Throttle parallelism*** *(int)*:
  Set by default at 3.
- **Renewal period** *(finite duration)*:
  Must be a valid finite duration.

5. Click on the save button.

You can update  or delete  the Intune PKCS Connector.

CAUTION	You won’t be able to delete an Intune PKCS Connector if it is referenced in any other configuration element.

9.5.3. Intune PKCS Profile

This section details how to configure the Intune PKCS Profile

Required By

Intune PKCS Scheduled Tasks

Prerequisites

PKI Connector

How to configure Intune PKCS Profile

1. Log in to Horizon Administration Interface.

2. Access Intune PKCS Profiles from the drawer or card: **Third Parties** > **Intune PKCS** > **Profiles**.

3. Click on  .

4. Fill the mandatory fields.

General

- **Name*** *(string input)*:
  Enter a meaningful connector name. It must be unique for each profile. Horizon uses the name to identify the profile.
- **Enabled*** *(boolean)*:
  Is the profile enabled or not. Set at true by default.
- **Max certificate per holder** *(int)*:
  When specified, define the maximum number of active certificates for a given Holder.
- **PKI Connector** *(select)*:
  Select a PKI connector previously created.
- **Intune PKCS Connector*** *(select)*:
  Select an Intune PKCS Connector previously created.

Crypto Policy

- **Private key escrowing** *(boolean)*:
  Is the private key escrowing. Set at false by default.
- **PKCS#12 Password generation mode*** *(select)*:
  Define if the PKCS#12 password is chosen by the user on the request (manual) or generate randomly (random).
- **Password policy for PKCS#12 password*** *(select)*:
  Select a password policy previously created.
- **Store encryption type** *(select)*:

Select from the list the encryption type. If unsure, leave on default "DES_AVERAGE".

- **Show PKCS#12 Password On Recover** *(boolean)*:
  Should the PKCS#12 password be displayed on recover. Activated with the private key escrowing. Set to false by default.
  - **Show PKCS#12 On Recover** *(boolean)*:
    Should the PKCS#12 be displayed on recover. Activated with the private key escrowing. Set to false by default.

Self Permissions

- **Revoke** *(boolean)*:
  Have the right to self revoke. Set by default at false.
- **Request Revoke** *(boolean)*:
  Have the right to self request revoke. Set by default at false.
- **Update** *(boolean)*:
  Have the right to self update. Set by default at false.
- **Request Update** *(boolean)*:
  Have the right to self request update. Set by default at false.
- **Recover** *(boolean)*:
  Have the right to self Recover the certificate. Set by default at false.
- **Request recover** *(boolean)*:
  Have the right to self request recover. Set by default at false.

Triggers

Intune PKCS profiles support the use of third-party triggers in the form of callbacks on specific events happening on the profile, giving a way to synchronize the third party repositories and Horizon.

- **Enrollment** *(select)*:
  Select the third party trigger(s) to call whenever a certificate is enrolled on this profile.
- **Revocation** *(select)*:
  Select the third party trigger(s) to call whenever a certificate gets revoked on this profile.
- **Expire** *(select)*:
  Select the third party trigger(s) to call whenever a certificate expires on this profile.

You can further configure the profile using the Common configuration profile and Notification tabs.

5. Click on the save button.

You can update  or delete  the Intune PKCS Profile.

CAUTION	You won't be able to delete an Intune PKCS Profile if it is referenced in any other configuration element.

9.5.4. Intune PKCS Scheduled Tasks

This section details how to schedule tasks that will run periodically on your Intune PKCS profiles.

Prerequisites

Connector Intune PKCS	Profile Intune PKCS

How to configure Intune PKCS Scheduled Tasks

1. Log in to Horizon Administration Interface.

2. Access Intune PKCS Scheduled Tasks from the drawer or card: **Third Parties** > **Intune PKCS** > **Scheduled Tasks**.

3. Click on  .

4. Fill the mandatory fields.

- **Enabled** *(boolean)*:
  Tells whether the Scheduled task should be enabled. Set by default at true.
- **Intune PKCS Profile*** *(select)*:
  Select an Intune PKCS profile previously created.
- **Target Connector*** *(select)*:
  Select an Intune PKCS connector previously created.
- **Cron scheduling** *(cron expression)*:
  By default set at every 5 hours.
- **Enroll?** *(boolean)*:
  If enabled, will enroll all certificate from the third party repository. Set to false by default.
- **Revoke?** *(boolean)*:
  If enabled, will revoke all certificate whose container was deleted from the third party repository. Set to false by default.
- **Renew?** *(boolean)*:
  If enabled, will renew all certificate who are about to expire. Set to false by default.
- **Dry run** *(boolean)*:
  If enabled, enroll, revocation and renewal actions will not be performed. Instead, a message will be logged, explaining what would have been done.

5. Click on the save button.

You can run  or update  or delete  the Schedules Tasks.

9.5.5. Intune PKCS Trigger

this section details how to configure the Triggers that will run automatically on your Intune PKCS connectors.

Prerequisites

Intune PKCS Connector

How to configure Intune PKCS Trigger

1. Log in to Horizon Administration Interface.

2. Access Intune PKCS Triggers from the drawer or card: **Third Parties** > **Intune PKCS** > **Triggers**.

3. Click on  .

4. Fill the mandatory fields.

- **Name*** *(string input)*:
  Enter a meaningful trigger name. It must be unique for each trigger. Horizon uses the name to identify the trigger.
- **Intune PKCS Connector*** *(select)*:
  Select an Intune PKCS connector previously created.
- **Retries in case of error** *(int)*:
  Number of times to retry to push the change on the Intune PKCS repository in case of error. Must be an integer between 1 and 15.

5. Click on the save button.

You can update  or delete  the Intune PKCS Trigger.

CAUTION	You won't be able to delete an Intune PKCS Trigger if it is referenced in any other configuration element.

9.6. jamf Pro

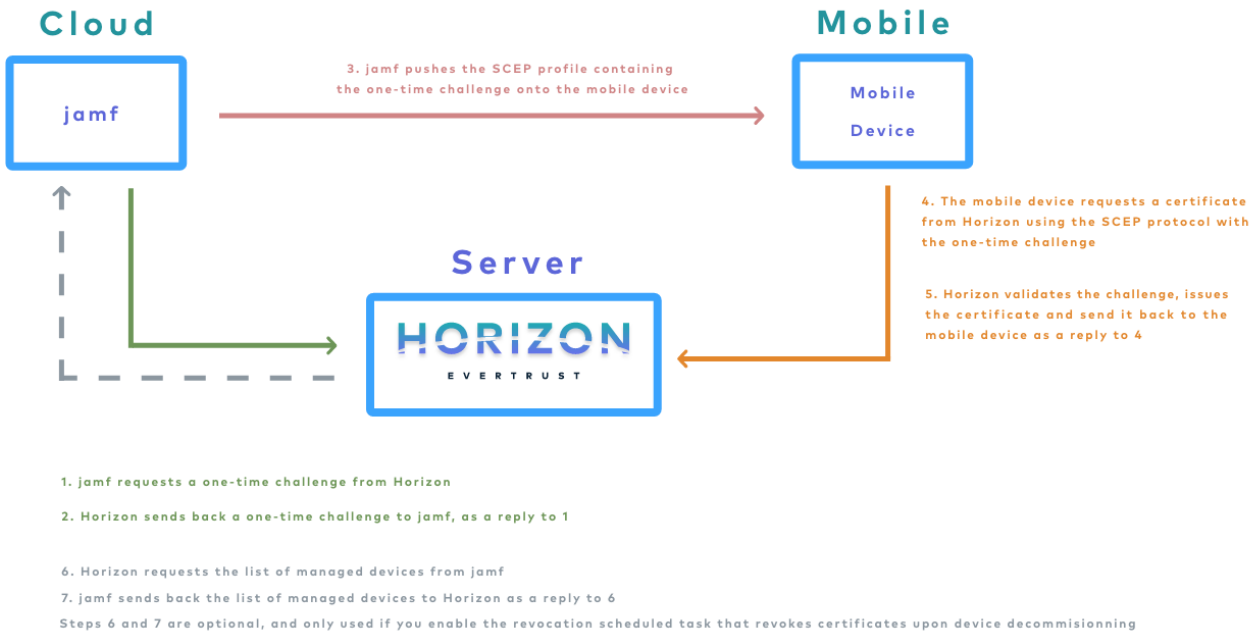
9.6.1. Introduction

This section details the jamf Pro integration with Horizon, used to enroll, renew and revoke certificates on jamf Pro managed devices.

This integration involves the following components:

- jamf Pro server or Cloud instance
- EverTrust Horizon

- Devices to be enrolled



The diagram displays these components as well as the various flows involved in an enrollment.

Finally, this integration will require to setup, on Horizon side, the following elements:

- a jamf Connector, which holds the configuration items required for Horizon to connect to jamf Pro
- a jamf Profile, which holds the configuration items specifying how Horizon should issue certificates for the specified jamf Connector
- a jamf Scheduled Task, which holds configuration items defining the scheduled task in charge of performing revocation upon decommissioning devices from jamf Pro. This is optional.

9.6.2. jamf Connector

This section details how to configure a jamf Connector.

Required By

jamf Profile

Prerequisites

On Horizon side, you might need to set up a Proxy used to reach jamf Pro, if necessary.

On jamf Pro side, it is required to create a technical user for Horizon, and give it **Auditor** rights, so that Horizon will be able to list the devices managed by jamf Pro and thus be able to trigger certificate revocation upon decommissioning. Please follow the steps from the jamf Pro documentation. After performing these steps, you will be given the following information, required later:

- a login
- a password

How to configure jamf Connector

1. Log in to Horizon Administration Interface.

2. Access jamf Connector from the drawer or card: **Third Parties** > **Jamf** > **Connectors**.

3. Click on  .

4. Fill the mandatory fields.

Connection

- **Name*** *(string input)*:
  Enter a meaningful connector name. It must be unique for each connector. Horizon uses the name to identify the connector.
- **Jamf endpoint URL*** *(string input)*:
  Enter the URL pointing to the jamf deployment or the jamf Cloud instance.
- **Login*** *(string input)*:
  Enter the username created for Horizon in jamf.
- **Password*** *(string input)*:
  Enter the password associated with aforementioned username.
- Proxy *(select)*:
  The HTTP/HTTPS proxy used to reach jamf Pro, if any.
- **Timeout** *(finite duration)*:
  Set by default at 10 seconds. Must be a valid finite duration.

Actors management

These configuration elements mainly define the number of authorized interactions with the remote service on a defined period. For example, one needs to ensure that the remote service will not be contacted more than 5 times per 3 seconds. *Throttle parallelism* defines the number of times and *Throttle duration* the period of time. Therefore, on the above example, throttle parallelism would be set to 5 and throttle duration would be set to 3 seconds.

- **Throttle duration*** *(finite duration)*:
  Set by default to 3 seconds. Must be a valid finite duration.
- **Throttle parallelism*** *(int)*:
  Set by default to 3.

5. Click on the save button.

You can update  or delete  the jamf Connector.

CAUTION	You won't be able to delete a jamf Connector if it is referenced in any other configuration element.

9.6.3. jamf Profile

This section details how to configure a jamf Profile

Prerequisites

jamf Connector	PKI Connector	SCEP Authorities

The SCEP Authority setup requires you to issue a certificate from the underlying PKI with the following characteristics:

- to issue certificates for iOS:
  - the issuing CA should be the same as the one that will issue certificates through the PKI Connector that will be linked to the jamf Profile
  - the certificate key usages must include **Digital Signature** and **Key Encipherment**
  - the certificate must be issued as PKCS#12 and then imported into Horizon
- to issue certificates for macOS:
  - the certificate should be self-signed
  - the certificate key usages must include **Digital Signature** and **Key Encipherment**
  - the certificate must be issued as PKCS#12 and then imported into Horizon

How to configure jamf Profile

1. Log in to Horizon Administration Interface.

2. Access jamf Profiles from the drawer or card: **Third Parties** > **Jamf** > **Profiles**.

3. Click on  .

4. Fill the mandatory fields.

General

- **Name*** *(string input)*:
  Enter a meaningful profile name. It must be unique for each profile. Horizon uses the name to identify the profile. As the name will be part of an URL, it is advised to use only lower case letters and dashes.
- **Enabled** *(boolean)*:
  Is the profile enabled or not. Set at true by default.
- **jamf Connector** *(select)*:
  Select a jamf connector previously created.

- **PKI connector*** *(select)*:
  Select a PKI Connector previously created.
- **Max certificate per holder** *(int)*:
  When specified, define the maximum number of active certificates for a given Holder.

Assets identification

- **DN field containing the device `UDID`*** *(select)*:
  Field used to retrieve the Device ID. The selected field must be set to `$UDID/$COMPUTERNAME` on jamf side, e.g. if you select "L", the configured Subject DN in the SCEP profile in jamf pro must then contain `L=$UDID` for iOS or `L=$COMPUTERNAME` for macOS devices. This allows to use the automated revocation upon device decommissioning feature.

SCEP protocol parameters

- **Mode*** *(select)*:
  Choose from the two modes RA or CA. To enroll certificates on **iOS** devices, select the **RA** mode. To enroll certificates on **macOS**, select the **CA** mode.
- **SCEP Authority*** *(select)*:
  Select a SCEP Authority previously created. See Prerequisites for details.
- **CAPS** *(select)*:
  Select one or many SCEP Capabilities from the list. If unsure, leave the default.
- **Encryption algorithm*** *(select)*:
  Select a SCEP Encryption Algorithm algorithms from the list. If unsure, leave the default.
- **Password policy** *(select)*:
  Choose from the password policy you might have previously created. If unsure, leave the default.

Renewal management

- **Renewal period** *(finite duration)*:
  Must be in valid finite duration.
- **Revocation on SCEP renew***: *(boolean)*
  Should the expiring certificate be revoked upon SCEP renewal. Set by default at false.
- **Revocation reason*** *(select)*:
  Select the reason from the list. Available only if revocation on SCEP renew at true.

Self Permissions

- **Revoke** *(boolean)*:
  Specify whether the certificate's owner is authorized to revoke the certificate with no validation workflow. Set to false by default.
- **Request Revoke** *(boolean)*:
  Specify whether the certificate's owner is authorized to request the revocation of the certificate. Set to false by default.

- **Update** *(boolean)*:
  Specify whether the certificate's owner is authorized to update the certificate with no validation workflow. Set to false by default.
- **Request Update** *(boolean)*:
  Specify whether the certificate's owner is authorized to request certificate's update. Set to false by default.

You can further configure the profile using the Common configuration profile and Notification tabs.

5. Click on the save button.

You can update  or delete  the jamf Profile .

CAUTION	You won't be able to delete a jamf Profile if it is referenced somewhere else.

Last Steps

The integration between jamf Pro and Horizon can be done in the following modes:

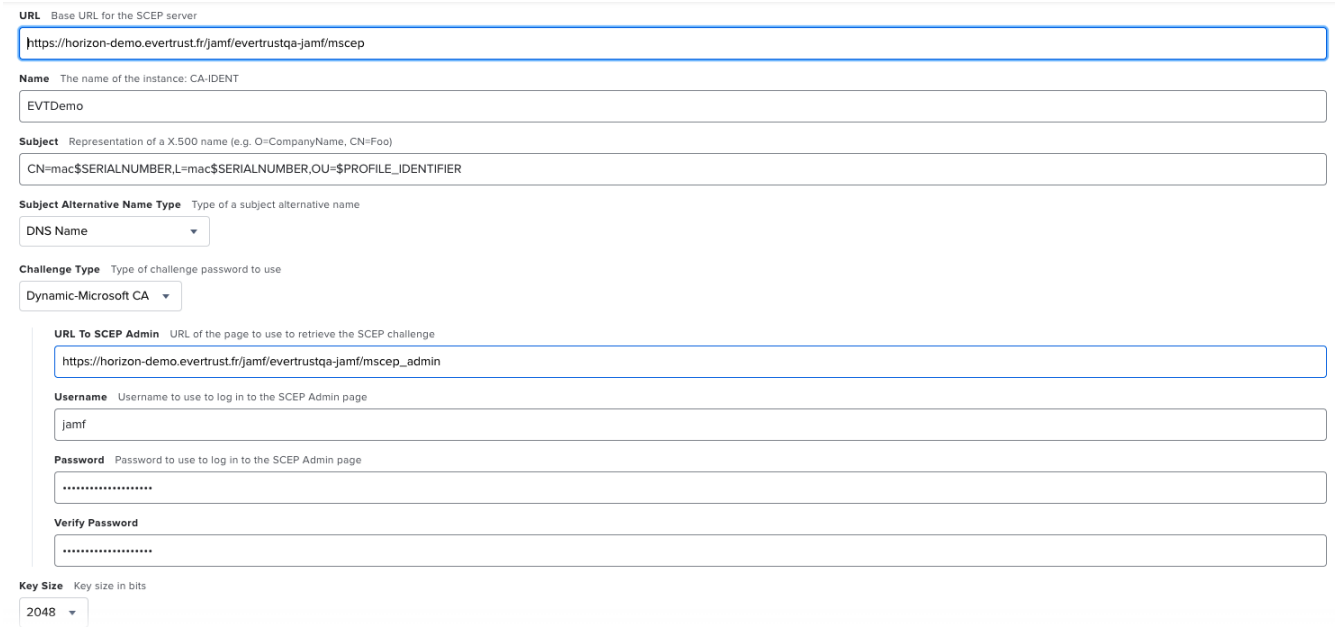
- jamf Pro SCEP Proxy mode
- iOS SCEP Profile
- macOS SCEP Profile
- macOS SCEP Profile with Proxy

In all these modes, the Challenge type to use on jamf Pro side is **Dynamic-Microsoft CA**, and you should point to the corresponding `mscep` and `mscep_admin` URI on Horizon side, that can be found in the jamf Profile after it has been created.

Jamf Pro SCEP Proxy mode

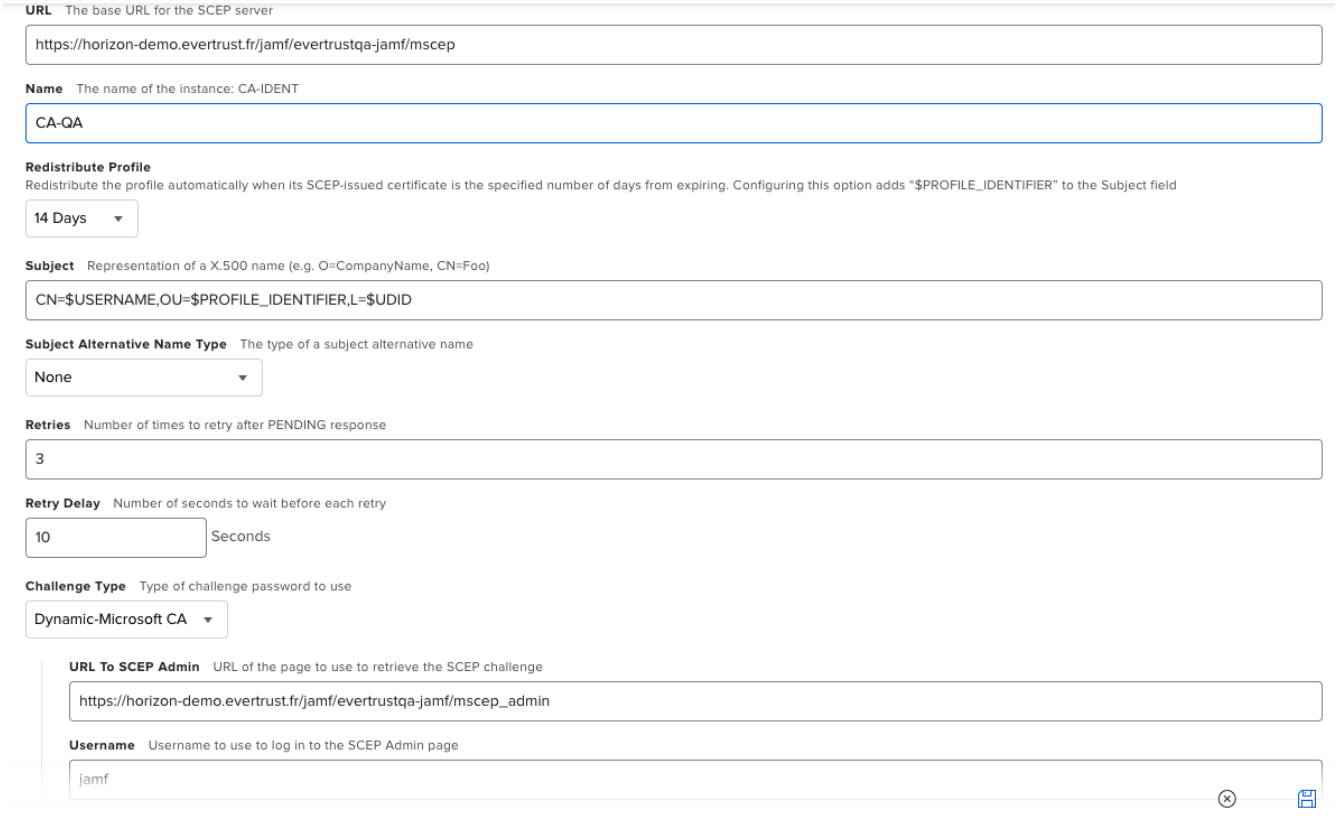
This mode requires to provide the SCEP Authority PKCS#12 to jamf Pro, so that it can be uploaded in the appropriate profile.

Other than that, the configuration looks like the following on Jamf Pro side:



iOS/macOS SCEP Profile

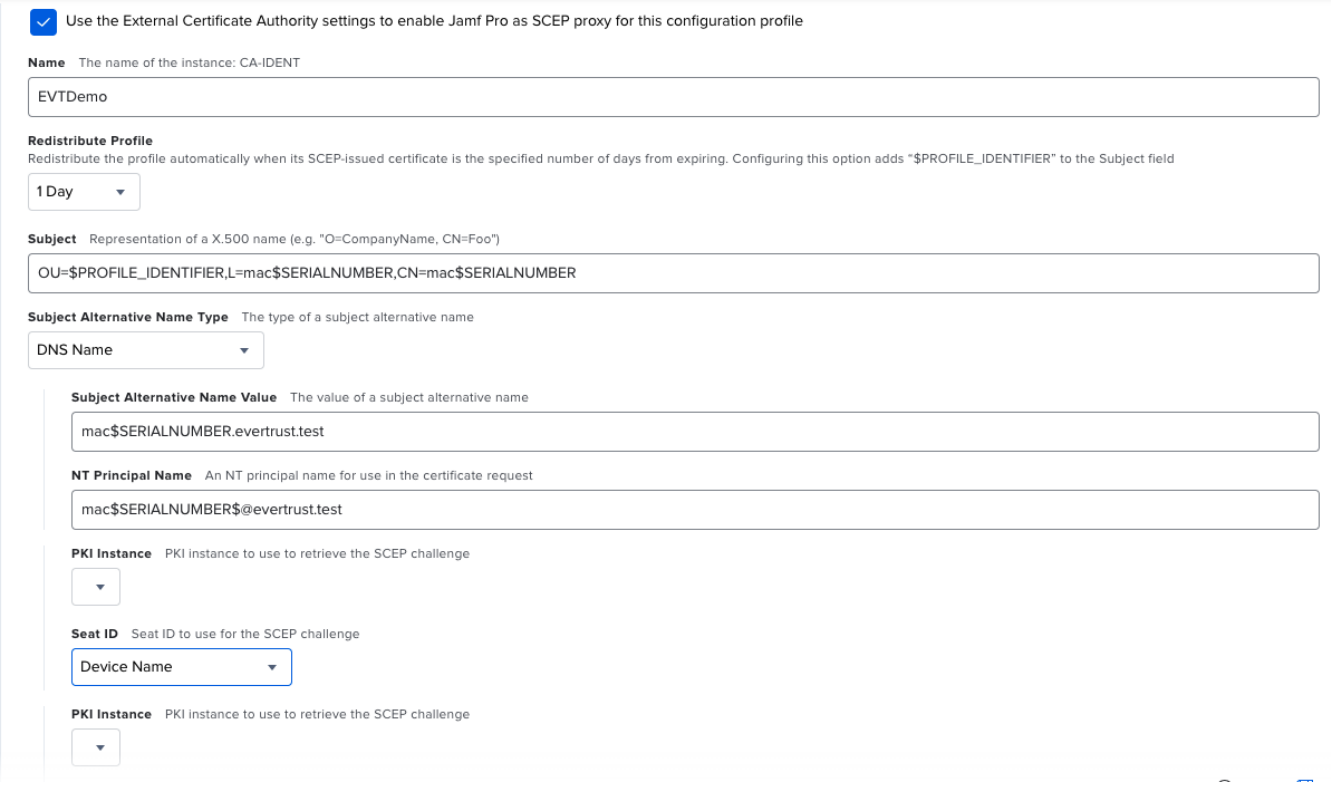
On jamf Pro side, the profile configuration looks like the following:



macOS SCEP Profile with Proxy

This mode requires:

1. to set up the SCEP Proxy mode on jamf Pro side
2. to configure a profile on jamf Pro side, that looks like the following:



9.6.4. jamf Scheduled Tasks

This section details how to schedule tasks that will run periodically on your jamf profiles, in order to manage automatic revocation upon device decommissioning.

Prerequisites

jamf Connector	jamf Profile

How to configure jamf Scheduled Tasks

1. Log in to Horizon Administration Interface.

2. Access jamf Scheduled Tasks from the drawer or card: **Third Parties** > **jamf** > **Scheduled Tasks**.

3. Click on ⊕ .

4. Fill the mandatory fields.

- **Enabled** *(boolean)*:
  Tells whether the Scheduled task should be enabled. Set by default at true.
- **jamf Profile*** *(select)*:
  Select a jamf profile previously created.
- **Target Connector*** *(select)*:
  Select an jamf connector previously created.
- **Cron scheduling** *(cron expression)*:
  Set to every 5 hours by default.

- **Revoke** *(boolean)*:
  Set to false by default. If true, Horizon will revoke any certificate associated to a device that has been deleted from Azure AD (and hence decommissioned).
- **Dry run** *(boolean)*:
  If enabled, revocation actions will not be performed. Instead, a message will be logged, explaining what would have been done.

5. Click on the save button.

You can run  or update  or delete  the Scheduled Tasks.

9.7. LDAP

9.7.1. Introduction

This section details the LDAP integration with Horizon, used to publish and unpublish certificates on LDAP.

The integration will require to set up the following elements (on Horizon side):

- an LDAP Connector, which holds the configuration items required by Horizon to connect to LDAP
- an LDAP Trigger, which holds the configuration items specifying how Horizon should publish/unpublish certificates for the specified LDAP connector

CAUTION	Only SMIME Certificates can be published

9.7.2. LDAP Connector

This section details how to configure an LDAP Connector.

Required By

LDAP trigger

Prerequisites

On the LDAP side, it is required to create a technical user with permissions to write in the LDAP sub DN, so that Horizon will be able to search by email, to publish and to unpublish certificates using that technical user. The following information will be required later:

- LDAP Hostname
- a login DN
- a password
- Base DN to publish SMIME certificates

How to configure LDAP Connector

1. Log in to Horizon Administration Interface.

2. Access LDAP Connector from the drawer or card: **Third Parties** > **LDAP** > **Connectors**.

3. Click on  .

4. Fill the mandatory fields.

Connection

- **Name*** *(string input)*:
  Enter a meaningful connector name. It must be unique for each connector. Horizon uses the name to identify the connector.
- **Hostname*** *(string input)*:
  Enter the URL pointing to LDAP.
- **Login DN*** *(string input)*:
  Enter the DN technical user created for Horizon.
- **Password*** *(string input)*:
  Enter the password associated with the login.
- **Base DN*** *(string input)*:
  Enter the Base DN where Horizon should publish the certificate.
- **Max stored certificates per holder*** *(int)*:
  When specified, define a maximum number of certificates stored in the third party.
- **Port** *(int)*:
  Enter the port where to reach the running LDAP instance (default values are 389 for LDAP and 636 for LDAPS).
- Proxy *(string input)*:
  The HTTP/HTTPS proxy used to reach LDAP, if any.
- **Timeout** *(finite duration)*:
  Set by default at 10 seconds. Must be a valid finite duration.

Assets identification

- **Filter*** *(string input)*:
  Enter the custom filter. By default, LDAP Identities are filtered by (objectclass=user). If you are using inetOrgPerson as type, you will have to manually set the following filter: (objectclass=inetOrgPerson).
- **Target LDAP publication attribute** *(string input)*:
  When specified, the certificate will be published on the specified attribute. In most LDAP applications you will have to set the field to: "userCertificate;binary" but in MSAD the field is already well managed.

Actors management

These configuration elements mainly define the number of authorized interactions with the remote service on a defined period. For example, one needs to ensure that the remote service will not be contacted more than 5 times per 3 seconds. *Throttle parallelism* defines the number of times and *Throttle duration* the period of time. Therefore, on the above example, throttle parallelism would be set to 5 and throttle duration would be set to 3 seconds.

- **Throttle duration*** *(finite duration)*:
  Set by default to 3 seconds. Must be a valid finite duration.
- **Throttle parallelism*** *(int)*:
  Set by default to 3.

5. Click on the save button.

You can update  or delete  the LDAP Connector.

CAUTION	You won't be able to delete a LDAP Connector if it is referenced in any other configuration element.

9.7.3. LDAP Triggers

Here is the section to manage the Triggers that will be used by profiles to publish or unpublish certificates into LDAP.

Prerequisites

LDAP Connector

How to configure LDAP trigger

1. Log in to Horizon Administration Interface.

2. Access LDAP triggers from the drawer or card: **Third Parties** > **LDAP** > **Triggers**.

3. Click on  .

4. Fill the mandatory fields.

- **Name*** *(string input)*:
  Enter a meaningful trigger name. It must be unique for each trigger. Horizon uses the name to identify the trigger.
- **LDAP Connector Certificate Publication*** *(select)*:
  Select an LDAP connector previously created.
- **Retries in case of error** *(int)*:
  Number of times to retry to push the change on the Intune PKCS repository in case of error. Must be an integer between 1 and 15.

5. Click on the save button.

You can run  or update  or delete  the trigger.

9.7.4. Integration of the third party to the WebRA

When having configured the connector, it is possible to automate its elements' lifecycle using the WebRA.

Automation using triggers

Triggers are a functionality of WebRA that allows to push lifecycle events into a third party whenever they occur on a WebRA profile.

1. Refer to the trigger documentation to create a trigger.

2. Create or modify the WebRA profile you wish to use the triggers on.

3. Go to the **Triggers** tab, then on **Certificate lifecycle triggers**

4. Chose which lifecycle events you wish to use triggers upon (enrollment, revocation, expiration)

5. Select one or more existing triggers from the menu (if several are selected, they will all be called whenever the selected event occurs)

6. Click on the Save button.

From now on, whenever a selected lifecycle event will occur on the configured WebRA profile, the trigger will be called and the and the certificate will be pushed into or removed from the third party container.

10. System configuration

10.1. SCEP Authorities

This section details how to configure SCEP Authorities.

The draft-nourse-scep-23 as well as RFC 8894 define how SCEP communications are secured. This involves using a SCEP Authority, which is a certificate and its associated private key, used to sign and encrypt communications between SCEP server and client.

Two setups are possible:

- the **CA mode** in which the SCEP Authority is a self-signed certificate. In that mode the SCEP server returns the self-signed certificate as `application/x-x509-ca-cert` when the client uses the `GetCaCert` call.
- the **RA mode** in which the SCEP Authority is a certificate signed by the CA that will issue certificates using the considered SCEP profile. In that mode, the SCEP server returns the SCEP Authority certificate and its issuing CA chain as `application/x-x509-ca-ra-cert` when the client uses the `GetCaCert` call.

Therefore, it is important in each SCEP or MDM Profile to align the SCEP mode with the characteristics of the SCEP Authority configured in the current section.

10.1.1. Prerequisites

- PKCS#12 containing the SCEP Authority certificate and private key. See above for explanation about the SCEP contents.

10.1.2. How to configure a SCEP Authority

1. Log in to Horizon Administration Interface.

2. Access SCEP Authorities from the drawer or card: **System** > **SCEP Authorities**.

3. Click on  .

4. Fill the following fields:

- **Name*** *(string input)*:
  Enter a meaningful SCEP Authority name;
- **PKCS#12*** *(import p12)*:
  PKCS#12 of the SCEP Authority;
- **PKCS#12 Password*** *(string input)*:
  Password of the aforementioned PKCS#12.

5. Click on the create button to save.

You can update  or delete  the SCEP Authority.

CAUTION	You won't be able to delete a SCEP Authority if it is referenced in any other configuration element.

10.2. Labels

This section details how to configure the labels. Labels are metadata used to store information provided by the en-users in Horizon database, associated to a given certificate, but not contained in the certificate.

You will be able to associate the labels created in this section with your profiles in order to enrich the certificates that will be issued from them.

10.2.1. How to configure a Label

1. Log in to Horizon Administration Interface.

2. Access Labels from the drawer or card: **System** > **Labels**.

3. Click on  .

4. Fill the following fields:

- **Name*** *(string input)*:
  Enter a meaningful Label name.
- Language* *(select)*:
  Please refer to Languages section to set up localized Label display name and description.
- **Display name** *(string input)*:
  The name that will be displayed for this label in the selected language. Optional.
- **Description** *(string input)*:
  The description of this label in the selected language. Optional.

5. Click on the create button to save.

You can update  or delete  Labels.

CAUTION	You won't be able to delete a Label if it is referenced in any other configuration element.

10.3. HTTP Proxy

In this section you will be able to set up HTTP Proxies. HTTP Proxies may be used by Horizon to establish connection to various services.

10.3.1. How to configure an HTTP Proxy

1. Log in to Horizon Administration Interface.

2. Access HTTP Proxy from the drawer or card: **System** > **HTTP Proxies**.

3. Click on  .

4. Fill the mandatory fields.

- **Name*** *(string input)*:
  Enter a meaningful HTTP Proxy name.
- **Host*** *(string input)*:
  The Hostname or IP Address of the HTTP/HTTPS proxy to use.
- **Port*** *(int)*:
  The Port of the HTTP/HTTPS proxy to use.

5. Click on the create button to save.

You can update  or delete  the HTTP Proxy.

CAUTION	You won't be able to delete an HTTP Proxy if it is referenced in any other configuration element.

11. Common configuration elements

11.1. Cron Expression

Cron expressions are composed of 6 required fields and one optional field separated by white spaces. The fields are respectively described as follows:

Field Name	Allowed Values	Allowed Special Character
Seconds	0-59	- * /
Minutes	0-59	- * /
Hours	0-23	- * /
Day-of-month	1-31	- * ? / L W
Month	1-12 or JAN-DEC	- *
Day-of-Week	1-7 or SUN-SAT	- * ? / L #
Year (Optional)	empty, 1970-2199	- * /

Special characters

- * ("all values") - used to select all values within a field. For example, "*" in the minute field means *every minute*.
- **?** ("no specific value") - useful when you need to specify something in one of the two fields in which the character is allowed, but not the other. For example, if I want my trigger to fire on a particular day of the month (say, the 10th), but don't care what day of the week that happens to be, I would put "10" in the day-of-month field, and "?" in the day-of-week field. See the examples below for clarification.
- **-** - used to specify ranges. For example, "10-12" in the hour field means "the hours 10, 11 and 12".
- **,** - used to specify additional values. For example, "MON,WED,FRI" in the day-of-week field means "the days Monday, Wednesday, and Friday".
- / - used to specify increments. For example, "0/15" in the seconds field means "the seconds 0, 15, 30, and 45". And "5/15" in the seconds field means "the seconds 5, 20, 35, and 50". You can also specify '/' after the '*' character - in this case '*' is equivalent to having '0' before the '/'. '1/3' in the day-of-month field means "fire every 3 days starting on the first day of the month".
- **L** ("last") - has different meaning in each of the two fields it is allowed into. For example, the value "L" in the day-of-month field means "the last day of the month" - day 31 for January, day 28 for February on non-leap years. If used in the day-of-week field by itself, it simply means "7" or "SAT". But if used in the day-of-week field after another value, it means "the last xxx day of the month" - for example "6L" means "the last Friday of the month". You can also specify an offset from the last day of the month, such as "L-3" which would mean the third-to-last day of the calendar month. When using the 'L' option, it is important not to specify lists, or ranges of values, as you'll get confusing/unexpected results.
- **W** ("weekday") - used to specify the weekday (Monday-Friday) nearest the given day. As an example, if you were to specify "15W" as the value for the day-of-month field, the meaning is:

"the nearest weekday to the 15th of the month". So if the 15th is a Saturday, the trigger will fire on Friday the 14th. If the 15th is a Sunday, the trigger will fire on Monday the 16th. If the 15th is a Tuesday, then it will fire on Tuesday the 15th. However if you specify "1W" as the value for day-of-month, and the 1st is a Saturday, the trigger will fire on Monday the 3rd, as it will not 'jump' over the boundary of a month's days. The 'W' character can only be specified when the day-of-month is a single day, not a range or list of days.

- # - used to specify "the nth" XXX day of the month. For example, the value of "6#3" in the day-of-week field means "the third Friday of the month" (day 6 = Friday and "#3" = the 3rd one in the month). Other examples: "2#1" = the first Monday of the month and "4#5" = the fifth Wednesday of the month. Note that if you specify "#5" and there is not 5 of the given day-of-week in the month, then no firing will occur that month.

NOTE	The 'L' and 'W' characters can also be combined in the day-of-month field to yield '`LW`', which translates to **"last weekday of the month"**.

11.2. Finite Duration

The format of a *Finite Duration* is `"<length><unit>"`, where:

- White space is allowed between the parts.
- Length is a positive integer without the "+" sign.
- Valid possible units are described in the below table:

Unit	Short name	Long names
DAYS	d	day days
HOURS	h	hour hours
MINUTES	m	min mins minute minutes
SECONDS	s	sec secs second seconds
MILLISECONDS	ms	milli millis millisecond milliseconds

For example, 10 seconds will be written as "10 s", "10s", "10 sec" or "10 seconds".

12. Reports

A report is a CSV file sent in a scheduled mail. The CSV content is managed by:

- *HCQL query (certificates), HRQL query (requests)*
- *CSV fields shown*

12.1. Prerequisites

You may need Teams.

12.2. How to configure Reports

1. Log in to Horizon Administration Interface.

2. Access Reports from the drawer or card: **Reports**.

3. Click on  .

4. Fill in the mandatory fields.

12.2.1. Details

- **Enabled** *(boolean)*:
  Tells whether the reporting task should be enabled. Set by default at true.
- **Name*** *(string input)*:
  Enter a meaningful report name. It must be unique.
- **Cron scheduling expression in Quartz format*** *(cron expression)*:
  Enter a Cron scheduling expression (in Quartz format). The default expression is built to run the task every hour.

12.2.2. Recipients

Click on  to add a recipient.

You can either target :

- A static (recipient): you will need to set a valid email address.
- A team contact: you will need to select one of the enabled teams.
- A team manager: you will need to select one of the enabled teams.

12.2.3. Email

- **From*** *(string input)*:

Enter the email address that will appear in the "From" field of the email.

- **Subject*** *(string input)*:
  Enter the subject of the email.
- **Body** *(string input)*:
  Enter the body of the email.
- **CSV file name** *(string input)*:
  Enter the name that will be given to the attached csv file.
- **Is HTML** *(boolean)*: (boolean):
  Sets whether the email body contains HTML code (true) or plain text (false). The default value is set to false.

12.2.4. HQL

- **HQL Type*** *(select)*:
  Either chose Certificate or Request. It will define the HQL Query type to set and the enabled CSV fields.
- **Query** *(string input or select)*:
  HCQL (Certificate) or HRQL (Request). You can select one of your saved queries.

12.2.5. CSV

You can select which fields will appear on the CSV file.

5. Click on the save button.

You can run , edit or delete the report .

13. Syslog Integration

Horizon is able to push its events (functional logs) to a syslog instance. This integration is pretty straightforward and can be implemented 2 ways :

13.1. Directly sending logs to your syslog server

1. Access the EverTrust Horizon server through SSH with an account with administrative privileges;

2. Using an editor like vi, open the *horizon-logback.xml* file located at **/opt/horizon/etc/horizon-logback.xml** ;

3. Edit the appender named "SYSLOG" to change the IP address for the syslogHost to redirect to your own syslog server. As an example, if your syslog server is on 192.168.1.2 and the Horizon logs must be processed by the LOCAL6 facility, the syslog appender should look like this :

```
<appender name="SYSLOG" class="ch.qos.logback.classic.net.SyslogAppender">
    <syslogHost>192.168.1.2</syslogHost>
    <facility>LOCAL6</facility>
    <suffixPattern>%msg%n</suffixPattern>
</appender>
```

4. Still in the **horizon-logback.xml** file, update the syslog logger and ensure that the log level is set to "INFO":

```
<logger name="syslog" level="INFO">
    <appender-ref ref="SYSLOG"/>
</logger>
```

5. Save your modifications and restart the Horizon service :

```
$ systemctl restart horizon
```

The functional logs from Horizon should now be received by your remote syslog server :

```
horizon {"code": "SERVICE-
STOP","details":[{"key":"horizonVersion","value":"2.3.4"},{"key":"message","value":"Se
rvice successfully
stopped"}],"module":"service","node":"horizon","timestamp":1674054152149,"status":"suc
cess"}
horizon {"code": "SERVICE-
START","details":[{"key":"horizonVersion","value":"2.3.4"},{"key":"message","value":"S
ervice successfully
started"}],"module":"service","node":"horizon","timestamp":1674054170567,"status":"suc
cess"}
```

13.2. Using the local syslog server for filtering and forwarding

Alternatively, you might want to use a local syslog instance to add grok filtering to your logs before forwarding them to your own syslog server. To do so, ensure that you have a syslog instance running (like **rsyslog**), then :

1. Access the EverTrust Horizon server through SSH with an account with administrative privileges;

2. With an editor like vi, edit the **/etc/rsyslog.d/horizon.conf** (or create it if it does not exist yet) to add this line :

```
local6.*                                                @REMOTE_SYSLOG_HOSTNAME
```

Don't forget to replace the *REMOTE_SYSLOG_HOSTNAME* to the IP or DNS name of your remote syslog server. As an example, if your syslog server is on 192.168.1.2, the line should look like this :

```
local6.*                                                @192.168.1.2
```

Note that you must set up your syslog host to accept UDP traffic on a specific port (here, we are going to use the default port which is 514) and catch the local6 facility logs, however the configuration of your own syslog host is out of the scope of this document.

3. Edit the */etc/rsyslog.conf* file to uncomment the module and input lines of the UDP section :

```
#module(load="imudp") # needs to be done just once
#input(type="imudp" port="514")
```

They should look like this after uncommenting :

```
module(load="imudp") # needs to be done just once
input(type="imudp" port="514")
```

4. Restart your syslog service :

```
$ systemctl restart rsyslog
```

The functional logs from Horizon should now be received by your remote syslog server, and you can add filtering on the */etc/rsyslog.d/horizon.conf* file before the logs actually get forwarded :

```
horizon {"code": "SERVICE-
STOP","details":[{"key":"horizonVersion","value":"2.3.4"},{"key":"message","value":"Se
rvice successfully
```

```
stopped"}],"module":"service","node":"horizon","timestamp":1674056069695,"status":"suc
cess"}
horizon {"code": "SERVICE-
START","details":[{"key":"horizonVersion","value":"2.3.4"},{"key":"message","value":"S
ervice successfully
started"}],"module":"service","node":"horizon","timestamp":1674056087880,"status":"suc
cess"}
```